(19)日本国特許庁（ＪＰ）　(12)公表特許公報（Ａ）　(11)特許出願公表番号
特表2001－508564
（P2001－508564A）
(43)公表日　平成13年６月26日(2001.6.26)

(51)Int.Cl.[7]	識別記号	ＦＩ	ﾃｰﾏｺｰﾄﾞ(参考)
Ｇ０６Ｆ　9/06	５５０	Ｇ０６Ｆ　9/06	５５０Ｚ

審査請求　未請求　予備審査請求　有　（全 47 頁）

(21)出願番号　特願平10－516593
(86)（22)出願日　平成９年９月29日(1997.9.29)
(85)翻訳文提出日　平成11年３月30日(1999.3.30)
(86)国際出願番号　ＰＣＴ／ＵＳ９７／１６６７５
(87)国際公開番号　ＷＯ９８／１４８７２
(87)国際公開日　平成10年４月９日(1998.4.9)
(31)優先権主張番号　０８／７２４，９４９
(32)優先日　平成８年10月２日(1996.10.2)
(33)優先権主張国　米国（ＵＳ）
(81)指定国　ＥＰ(ＡＴ，ＢＥ，ＣＨ，ＤＥ，ＤＫ，ＥＳ，ＦＩ，ＦＲ，ＧＢ，ＧＲ，ＩＥ，ＩＴ，ＬＵ，ＭＣ，ＮＬ，ＰＴ，ＳＥ)，ＡＵ，ＢＲ，ＣＡ，ＣＮ，ＩＬ，ＪＰ，ＮＯ

(71)出願人　トレンド　マイクロ，インコーポレーテッド
台湾　タイペイ，チンシャン　サウス　ロード，セクション　２，ナンバー　218，エスエフ
(71)出願人　チェン，エヴァ　ワイ．
アメリカ合衆国　カリフォルニア州　95014　クペルティーノ，オレンジ　アヴェニュー　10408
(74)代理人　弁理士　内原　晋

最終頁に続く

(54)【発明の名称】　マクロ中のウイルスの検出および除去のためのシステム、装置および方法

(57)【要約】
マクロの中のウイルスの検出および除去を開示している。マクロウイルス検出モジュール（２０６）はマクロ位置特定および復号化モジュール（３０２）と、マクロウイルス走査モジュール（３０４）と、マクロ処置モジュール（３０６）と、ファイル処置モジュール（３１０）と、ウイルス情報モジュール（３０８）とを含む。マクロ位置特定および復号化モジュール（３０２）は、対象ファイルがマクロを含むか否かを判定し、マクロ有りの場合にそのマクロを位置特定して復号化ずみマクロを生ずるように復号化する。マクロウイルス走査モジュール（３０４）は復号化ずみマクロにアクセスし、そのマクロがウイルスを含むか否かを判定するようにそのマクロを走査する。未知のマクロウイルスの検出は、マクロウイルス走査モジュール（３０４）が、ウイルス情報モジュール（３０８）からの命令識別子組含有比較データの取込みおよび前記命令識別子対応の容疑命令組合せの前記復号化ずみマクロにおける含有の有無の判定に基づいて行う。マクロ処置モジュール（３１０）は前記比較データを用いて前記復号化ずみマクロの中で容疑命令

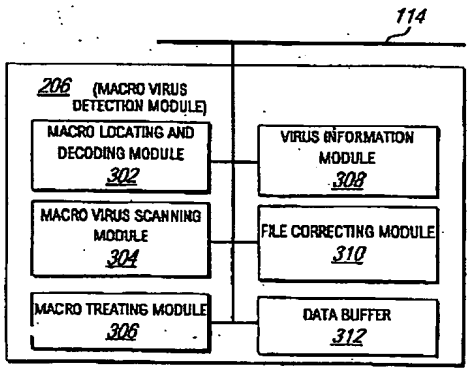


FIG. 3

【特許請求の範囲】

１．プロセッサおよびメモリ装置を含むコンピュータシステムにおいてマクロの中のウイルスを検出する方法であって、

ウイルス検出のための情報を含む比較データを得る過程と、

マクロを読み込む過程と、

復号化ずみマクロを生ずるように前記マクロを復号化する過程と、

前記復号化ずみマクロを前記比較データと比較することによってウイルスについて前記復号化ずみマクロを走査する過程と

を含む方法。

２．前記復号化ずみマクロを走査する過程がそのマクロの前記ウイルスによる汚染を示す場合に処置ずみマクロを生ずるように前記マクロから前記ウイルスを除去する過程

をさらに含む請求項１記載の方法。

３．前記マクロを読み込む過程が、

対象ファイルがテンプレートファイルであるか否かを判定する過程と、

前記対象ファイルがテンプレートファイルでない場合にその対象ファイルが埋込みマクロを含むか否かを判定する過程と、

前記対象ファイルがテンプレートファイルを含む場合に前記埋込みマクロを位置特定する過程と

を含む請求項１記載の方法。

４．前記比較データが第１の容疑命令識別子および第２の容疑命令識別子を含む請求項１記載の方法。

５．前記ウイルスについて前記復号化ずみデータを走査する過程が、

前記復号化ずみマクロが前記第１の容疑命令識別子に対応する第１の部分を含むか否かを判定する過程と、

前記復号化ずみマクロが前記第２の容疑命令識別子に対応する第２の部分を含むか否かを判定する過程と、

前記復号化ずみマクロが前記第１および第２の部分を含む場合に前記復号化ず

みマクロが前記ウイルスを含むと判定する過程と
を含む請求項４記載の方法。
６．前記第１の容疑命令識別子がマクロウイルスイネーブル化命令を検出する請求項５記載の方法。
７．前記第２の容疑命令識別子がマクロウイルス複製命令を検出する請求項６記載の方法。
８．前記ウイルスを除去する過程が、
前記第１の容疑命令識別子に対応する第１のマクロ命令を前記復号化ずみマクロの中で位置特定する過程と、
前記第１の容疑マクロ命令を除去する過程と
を含む請求項２記載の方法。
９．前記処置されたマクロの完全無欠性を検証する過程と、
前記処置されたマクロの完全無欠性検証に応じて、対象ファイルの中の前記汚染マクロを修復ずみマクロに置換する過程と
をさらに含む請求項８記載の方法。
１０．前記第１の容疑マクロ命令を除去する過程が前記第１の容疑命令を非汚染命令に置換する過程を含む請求項８記載の方法。
１１．前記ウイルスを除去する過程が、
前記第２の容疑命令識別子に対応する第２の容疑マクロ命令を前記復号化ずみマクロの中で位置特定する過程と、

処置ずみマクロを生ずるように前記第２の容疑マクロ命令を前記復号化ずみマクロから除去する過程と
を含む請求項８記載の方法。
１２．前記比較データが複数の容疑命令識別子組を含む請求項１記載の方法。
１３．第１の容疑命令識別子組がストリング73 CB 00 0C 6C 01 00および67 C2 80を含む請求項１２記載の方法。
１４．第２の容疑命令識別子組がストリング73 CB 00 0C 6C 01 00および64 6F 02 67 DE 00 73 87 01 12 73 7Fを含み、第３の容疑命令識別子組がストリング7

3 CB 00 0C 6C 01 00および6D 61 63 72 6F 73 76 08を含み、第４の容疑命令識別子組がストリング12 6C 01 00および64 67 C2 80 6A 0F 47を含み、第５の容疑命令識別子組がストリング79 7C 66 6F 72 6D 61 74 20 63 6Aおよび80 05 6A 07 43 4F 4Dを含む請求項１３記載の方法。

１５．プロセッサおよびメモリ装置を含むコンピュータシステムにおいてマクロ中のウイルスを検出する方法であって、

マクロ読み込む過程と、

第１の容疑命令識別子および第２の容疑命令識別子を含み、ウイルスを検出するための比較データを得る過程と、

前記第１の容疑命令識別子に対応する第１の部分を前記マクロが含むか否かを判定するように前記マクロを走査する過程と、

前記第２の容疑命令識別子に対応する第２の部分を前記マクロが含むか否かを判定するように前記マクロを走査する過程と、

前記マクロが前記第１および第２の部分を含んでいた場合に前記マクロが前記ウイルスで汚染されていたと判定する過程と

を含む方法。

１６．前記マクロが前記第１および第２の部分を含むと判定された場合に処置ずみマクロを生ずるように前記マクロを処置する過程

をさらに含む請求項１５記載の方法。

１７．前記マクロを処置する過程が、

前記第１の容疑命令識別子に対応する第１のマクロ命令を前記汚染マクロの中で位置特定する過程と、

前記汚染マクロを修復するように前記第１のマクロ命令を前記汚染マクロから除去する過程と

を含む請求項１６記載の方法。

１８．前記マクロを処置する過程が、

前記第２の容疑命令識別子に対応する第２のマクロ命令を前記汚染マクロの中で位置特定する過程と、

前記汚染マクロを修復するように前記第２のマクロ命令を前記汚染マクロから除去する過程と

を含む請求項１７記載の方法。

１９．前記マクロを読み込む過程が、

対象ファイルにアクセスする過程と、

前記対象ファイルがテンプレートファイルであるかどうかを判定する過程と、

そのファイルがテンプレートファイルであった場合にそのファイルが埋込みマクロを含むか否かを判定する過程と、

そのファイルが埋込みマクロを含む場合にその埋込みマクロを位置特定する過程と

を含む請求項１５記載の方法。

２０．前記第１の容疑命令識別子がストリング73 CB 00 0C 6C 01 00を含み、前記第２の容疑命令識別子がストリング67 C2 80を含む請求項１５記載の方法。

２１．前記比較データが複数の容疑命令識別子を含む請求項１５記載の方法。

２２．第１の容疑命令識別子組がストリング73 CB 00 0C 6C 01および67 C2 82を含み、第２の容疑命令識別子組がストリング73 CB 00 0C 6C 01 00および64 6F 02 67 DE 00 73 87 01 12 73 7Fを含み、第３の容疑命令識別子組がストリング73 CB 00 0C 6C 01 00および6D 61 63 72 6F 73 76 08を含み、第４の容疑命令識別子組がストリング12 6C 01 00および64 67 C2 80 6A 0F 47を含み、第５の容疑命令識別子組がストリング79 7C 66 6F 72 6D 61 74 20 63 6Aおよび80 05 6A 07 43 4F 4Dを含む請求項２１記載の方法。

２３．対象ファイルにアクセスする過程と、

前記対象ファイルの中で前記マクロを位置特定する過程と、

前記マクロを前記対象ファイルから除去する過程と、

修復ずみファイルを生ずるように前記処置されたマクロを前記対象ファイルに加える過程と

をさらに含む請求項１５記載の方法。

２４．マクロの中のウイルスを検出する装置であって、

第１の容疑命令識別子および第２の容疑命令識別子を含みウイルスを検出するための比較データを蓄積するウイルス情報モジュールと、

前記ウイルス情報モジュールと信号授受して、前記比較データを受けるとともに、前記マクロが前記第１の容疑命令識別子対応の第１の部分および前記第２の容疑命令識別子対応の第２の部分を含むか否かを判定するように前記マクロを走査するマクロウイルス走査モジュールと

を含む装置。

２５．前記マクロウイルス走査モジュールと信号授受して、対象ファイルにアクセスし、その対象ファイルがテンプレートファイルであるか否かを判定し、その対象ファイルが埋込みマクロを含むか否かを判定し、復号化ずみマクロを生ずるようにそのマクロを復号化するマクロ位置特定および復号化モジュール

をさらに含む請求項２４記載の装置。

２６．前記ウイルス情報モジュールと信号授受して、前記復号化ずみマクロにアクセスし前記第１の容疑命令識別子対応の第１のマクロ命令および前記第２の容疑命令識別子対応の第２のマクロ命令を除去し処置ずみマクロを生ずるマクロ処置モジュール

をさらに含む請求項２５記載の装置。

２７．前記マクロ処置モジュールと信号授受して、前記対象ファイルにアクセスし前記対象ファイルの中でマクロを位置特定しその対象ファイルからそのマクロを除去し前記処置ずみマクロを前記対象ファイルに加えて訂正ずみファイルを生ずるファイル訂正モジュール

をさらに含む請求項２６記載の装置。

２８．前記第１の命令識別子がストリング73 CB 00 0C 6C 01 00を含み、前記第２の命令識別子がストリング67 C2 80を含む請求項２７記載の方法。

２９．前記比較データが複数の容疑命令識別子組を含む請求項２７記載の方法。

３０．第１の容疑命令識別子組がストリング73 CB 00 0C 6C 01 00および67 C2 82を含み、第２の容疑命令識別子組がストリング73 CB 00 0C 6C 01 00および64 6F 02 67 DE 00 73 87 01 12 73 7Fを含み、第３の容疑命令識別子組がストリ

ング73 CB 00 0C 6C 01 00および6D 61 63 72 6F 73 76 08を含み、第４の容疑命令識別子組がストリング12 6C 01 00および64 67 C2 80 6A 0F 47を含み、第５の容疑命令識別子組がストリング79 7C 66 6F 72 6D 61 74 20 63 6Aおよび80 05 6A 07 43 4F 4Dを含む請求項２９記載の方法。

３１．マクロの中のウイルスを検出する装置であって、

第１の容疑命令識別子および第２の容疑命令識別子を含む比較データをウイルス検出用に得る手段と、

前記第１の容疑命令識別子対応の第１の部分をマクロが含むか否かを判定するように前記マクロを走査する手段と、

前記第２の容疑命令識別子対応の第２の部分をマクロが含むか否かを判定するように前記マクロを走査する手段と、

前記第１および第２の部分を含んでいた場合に前記マクロがウイルスに汚染されていると判定する手段と

を含む装置。

３２．前記第１の容疑命令識別子および前記第２の容疑命令識別子にそれぞれ対応する第１のマクロ命令および第２のマクロ命令を前記マクロの中で位置特定する手段と、

処置ずみマクロを生ずるように前記第１のマクロ命令および前記第２のマクロ命令を前記マクロから除去する手段と

をさらに含む請求項３１記載の装置。

３３．対象ファイルにアクセスしその対象ファイルがマクロを含むか否かを判定する手段をさらに含む請求項３２記載の装置。

３４．前記対象ファイルにアクセスする手段と、前記マクロを前記対象ファイルから除去する手段と、訂正ずみファイルを生ずるように前記処置ずみマクロを前記対象ファイルに加える手段とを含むファイル訂正手段

をさらに含む請求項３３記載の装置。

３５．マクロの中のウイルスを検出するシステムであって、

第１の容疑命令識別子および第２の容疑命令識別子を含むウイルス検出用比較

データおよびルーチンを蓄積するメモリ装置と、

前記メモリ装置と信号授受し、前記マクロが前記第１の容疑命令識別子対応の第１の部分および前記第２の容疑命令識別子対応の第２の部分を含むか否かを判定するように前記比較データを受けるとともに前記マクロを走査するプロセッサと

を含むシステム。

【発明の詳細な説明】

マクロ中のウイルスの検出および除去のための システム、装置および方法

発明の背景

発明の技術分野

この発明は概括的にはコンピュータファイルの中のウイルスの検出および除去に関する。

関連技術の説明

コンピュータ利用の普及および莫大な数のコンピュータ相互間の通信の拡大はコンピュータウイルスの広がりを著しく容易にし助長してきた。コンピュータウイルスはコンピュータプログラムの中に埋め込まれたコードの種々の部分に見出される。ウイルスに感染したプログラムが実行されると、それらコード部分が活性化されて、コンピュータシステムに意図しない、場合によっては有害な動作を生じさせる。

ウイルスの検出は通常シグネチャ走査手法を用いて行う。既知のウイルスのために、ウイルス検出に使用可能な指紋同等のストリングまたはシグネチャが長くなった。シグネチャ走査では実行可能なファイルシーケンスを走査して、ウイルスとして既知のストリングに合致する延長ストリングを含むか否かを調べる。その実行可能なファイルの中に上記シグネチャまたはストリングが見出されると、積極的ウイルス判定を行う。既知のパターンとのマッチングを伴うので、パターンの特定されていないウイルスに対してはシグネチャ走査手法はほとんど役に立たない。とくに、新しい未知のウイルスの種類の検出はシグネチャ走査手法には全く不可能であり、複製の際に多様な形状および形式を意図的にとる突然変異性ウイルスに対しては十分な保護を提供できない。また、実行可能なファイル（例えば拡張子.exeまたは.comつきのファイル）も通常走査されるので、それらファイルの中にないウイルスは検査されず、したがってシグネチャ走査では検出されない。

多数のアプリケーションプログラムが、動作の長いシーケンスまたは反復シー

ケンスの自動的遂行のためのマクロの使用をサポートしている。マクロは、メニュー選択やキー押下や蓄積され名称またはキーの割当てを受けるコマンドなどの一連の命令である。マクロは、アプリケーションプログラムによってキーの押下げまたはマクロ名称の呼出しに応答して起動できる。アプリケーションデータファイルに埋め込まれていて、ユーザから隠れた状態に留まるマクロもある。また、マクロはユーザからの入力なしに自動的に実行するようにすることもできる。すなわち、ユーザに知られる必要がなくユーザによる始動を必要としないマクロがアプリケーションデータファイルなどのファイルに常駐することがあり得る。ある種のウイルスはマクロに常駐し、マクロ命令を用いて想定外の有害な動作を行う。それらのウイルスをマクロウイルスと呼ぶ。マクロウイルスの一つの問題は、実行可能なファイルには通常は常駐しないので実行可能なファイルスキャナーを避けることである。また、マクロウイルスはアプリケーションデータファイルなどのファイルの中に隠されたり埋め込まれたりされ得るので、検出を免れる。さらに、マクロプログラム用言語の使い方を知っているコンピュータユーザは多数にのぼるので、マクロウイルスの数と多様性は極めて大きい。したがって、マクロの中のウイルスの検出にシグネチャ走査手法を用いても、多数の未知のマクロウイルスがあるので効果がない。また、網羅的シグネチャスキャナーを使えたとしても、新たな未知のマクロウイルスの発生が継続するのでそのスキャナーはすぐ陳腐化する。

慣用のウイルス除去手法もウイルス感染マクロの処置には不十分である。それら慣用の手法は特定の既知のウイルスを探索し、その探索により検出された特定のウイルスに応じて特定の訂正手法を適用する。その訂正手法は、未知のマクロウイルスの莫大な糸列のために、ウイルス感染マクロの処置にはあまり効果がない。未知のマクロを検出しても、ウイルス感染マクロ含有のファイルを消去するだけでは有効な解決方法にはならない。ウイルス感染マクロの中にユーザが保持したい正常動作が含まれていることが多いからである。したがって、マクロからウイルス、とくに未知のウイルスを選択的に除去して、その後で使用可能な感染なしの訂正ずみファイルを生ずる必要がある。

もう一つの問題はウイルスもそれらウイルスの検出に必要な情報も常に変動し

ていることである。したがって、ウイルス検出情報の更新の容易なウイルス検出方法およびウイルス検出装置が必要である。容易に改変可能な未知マクロウイルス検出情報がとくに必要である。

上述のとおり、マクロに常駐するウイルスの検出が必要である。さらに、未知のマクロウイルスを検出し、マクロウイルスを選択的に清浄化し、マクロウイルス検出情報を簡便に更新する必要がある。

発明の概要

この発明はマクロからウイルスを検出し除去するシステム、装置および方法によって従来技術の制約と欠点を解消する。

この発明によると、マクロウイルス検出モジュールはマクロ位置特定および復号化モジュールと、マクロウイルス走査モジュールと、マクロ処置モジュールと、ウイルス情報モジュールと、ファイル訂正モジュールと、データバッファとを含む。マクロウイルス検出モジュールのコンフィギュレーション設定にしたがって一つのファイルをウイルス検出の対象にし、データバッファにコピーして分析に備える。マクロ位置特定および復号化モジュールでそのファイルを試験して、それがテンプレートファイルか否かを判定する。テンプレートファイルと判定した場合は、そのテンプレートファイルの中のマクロはすべて位置特定して復号化する。対象ファイルがテンプレートファイルでない場合は、マクロ位置特定および復号化モジュールでその対象ファイルを調べ、埋込みマクロを含むか否かを判定し、その位置特定を行い復号化する。復号化されたマクロはデータバッファに蓄積する。

マクロ走査モジュールはこのマクロ位置特定および復号化モジュールおよびデータバッファと信号授受し、したがって復号化ずみのマクロにアクセスしてウイルス走査に備える。マクロウイルス走査モジュールはマクロウイルス情報モジュールとも信号授受する。マクロウイルス情報モジュールは、マクロ中の既知および未知のウイルスの検出のためにマクロウイルス走査モジュールが用いる情報を含む。上記復号化ずみのマクロをまず既知のウイルスの探索のために走査する。既知のウイルスが検出された場合はその復号化ずみマクロは感染のフラグで表示する。その復号化ずみマクロ、そのフラグ、およびその復号化ずみマクロをその

マクロ中で検出された既知のマクロに関連づける情報をデータバッファに蓄積す

る。上述のとおり感染マクロおよびその感染マクロの常駐するファイルはマクロ処置モジュールおよびファイル訂正モジュールによって適切に処置され訂正される。

既知のウイルスが検出されなかった場合は、復号化ずみマクロが未知のウイルスが含んでいるか否かをマクロウイルス走査モジュールが判定する。マクロウイルス走査モジュールはウイルス情報モジュール中に蓄積されている比較データを用いて未知のマクロウイルスを検出する。この比較データは、マクロ中の容疑命令の組合せの検出に用いる情報を含む。比較データの組の好例は第１および第２の容疑命令特定コードを含む。マクロウイルス走査モジュールは、それら第１および第２の容疑命令が両方ともマクロに含まれていた場合にそのマクロがウイルスを含んでいると判定する。未知のウイルスが検出された場合は、その未知のウイルスの検出に導く容疑命令特定コードの組にしたがって、そのマクロを汚染マクロとフラグ表示する。既知のウイルスの検出の場合と同様に、明確な検出に導いた情報を汚染マクロとともにデータバッファに蓄積し、マクロ処置モジュールおよびファイル訂正モジュールが汚染マクロを適切に処置し訂正するようにする。特定のシーケンス状のシグネチャでなく容疑命令の組合せを探索するので、マクロウイルス走査モジュールで未知のウイルスが検出される。既知のウイルスおよび未知のウイルスの検出のための情報は別のモジュールに常駐しているので容易に更新される。

マクロ処置モジュールはマクロウイルス走査モジュールおよびデータバッファと信号授受し、それによって検出ウイルス関連の情報を得る。マクロ処置モジュールはマクロからウイルスを除去して清浄化ずみマクロまたはウイルス除去ずみマクロを発生し、汚染マクロ含有のファイルをファイル処置モジュールで補修または訂正できるようにする。マクロ処置モジュールはデータバッファ内の復号化ずみマクロにアクセスし、既知のウイルスによる汚染のフラグ表示があるか否かを判定する。そのマクロに既知のウイルスによる汚染のフラグ表示がある場合は、その既知のウイルスをそのマクロから除去する。そのマクロが既知のウイルス

に汚染されていない場合は、マクロ処置モジュールがマクロ中の未知のウイルスの検出のための命令特定コードの組を用いてマクロを処置する。このマクロ処置モジュールは、マクロウイルス走査モジュールおよびデータバッファから、復号化

ずみマクロおよびウイルスの存在に関する情報を受ける。復号化ずみマクロの中の容疑命令は命令特定コードを用いて特定し位置特定する。次に、好ましくは非感染命令で置換することによって汚染マクロから容疑命令を除去し、清浄化ずみまたはウイルス除去ずみの処置ずみマクロを生ずる。この処置ずみマクロはデータバッファに蓄積してファイル訂正モジュールによるアクセスに備える。この処置ずみマクロの完全無欠性を検査し、検査結果に応じてその有効性をフラグ表示する。マクロの完全無欠性がマクロ処置の完了の時点で維持されている場合は有効のフラグ表示をする。完全無欠性が維持されていない場合はフラグ表示しない。

ファイル訂正モジュールは、マクロ位置特定および復号化モジュール、マクロウイルス走査モジュール、マクロ処置モジュール、データバッファおよびウイルス情報モジュールと信号授受する。処置ずみのマクロおよび汚染マクロ含有の対象ファイルに関する情報はデータバッファ内でアクセスを受ける。ファイル訂正モジュールはもとの形の対象ファイルにアクセスし、対象ファイルのコピーをデータバッファに蓄積する。対象ファイルのコピーは汚染マクロを含む。マクロ有効性フラグ表示がない場合は、処置ずみマクロを汚染マクロ置換に用いることはせず、対象ファイル消去など代替の訂正動作を行わせ、汚染マクロ含有ファイルの存在のユーザへの通知、または対象ファイルからの汚染ファイルの除去およびマクロなしバージョンへの対象ファイルの置換を行う。マクロ有効性のフラグ表示がある場合は、ファイル訂正モジュールが汚染マクロを処置ずみマクロに置換することによって対象ファイルを訂正する。汚染マクロを置換するために、ファイル訂正モジュールは汚染マクロを位置特定して対象ファイルから除去し、そのマクロ抜きの対象ファイルのバージョンがデータバッファに蓄積されるようにする。次に処置ずみマクロを前記マクロ抜きの対象ファイルのバージョンに加えて

訂正ずみファイルを生ずる。この訂正ずみファイルを対象ファイル（もとの位置の）の置換に用いる。したがって、未知のウイルスはマクロから除去され、そのようなマクロを含むファイルは正しい機能を保持するよう修正される。

図面の簡単な説明

この発明の上記および上記以外の詳細な特定の特徴は添付図面を参照した次の説明により詳細に開示する。

図１はこの発明によるマクロウイルス検出装置を含むコンピュータシステムを図解するブロック図である。

図２はこの発明によるメモリ装置の好ましい実施例を図解するブロック図である。

図３はこの発明によるマクロウイルス検出モジュールの好ましい実施例を図解するブロック図である。

図４はこの発明によるマクロウイルス検出および訂正方法を図解する流れ図である。

図５はこの発明によるマクロの位置特定および復号化方法を図解する流れ図である。

図６はこの発明によるウイルス探索用マクロ走査方法を図解する流れ図である。

図７はこの発明によるマクロ処置方法を図解する流れ図である。

図８はこの発明によるファイル訂正方法を図解する流れ図である。

図９はマクロウイルスの検出に用いる比較データの組の好例を含む表である。

発明の詳細な説明

図１を参照すると、この発明によって構成したコンピュータシステム１００は中央処理装置（ＣＰＵ）１０４、表示装置１０２、メモリ装置１０６、入力装置１０８、データ蓄積装置１１０および通信ユニット１１２を含む。ＣＰＵ１０４は、パーソナルコンピュータの場合のようにフォン ノイマン アーキテクチャなど慣用のアーキテクチャでバス１１４によって表示装置１０２、メモリ装置１０６、入力装置１０８、データ蓄積装置１１０および通信ユニット１１２に接続す

る。ＣＰＵ１０４はカリフォルニア州サンタクララのインテル社から市販されているPentiumなどのマイクロプロセッサ、表示装置１０２はビデオモニタ、メモリ装置１０６はランダムアクセスメモリ（RAM）、入力装置１０８はキーボードおよびマウス、データ蓄積装置１１０はハードディスク装置、および通信ユニット１１２は他システムとの信号授受を容易にするモデムなどの装置でそれぞれ構成するのが好ましい。

上記以外の多様なコンピュータシステム構成が利用可能であり、この発明はそれら構成のいずれを用いるかに制約されない。例えば、ＣＰＵ１０４にはモトローラ社から市販されている代替のプロセッサを用いることができ、メモリ装置１０６は読出専用メモリ（ROM）またはRAMおよびROMの組合せでも構成できる。また、

システム１００をネットワークインタフェース（図示してない）を通ずるなどして他のコンピュータシステムに接続することもできる。また、コンピュータシステム１００はミニコンピュータまたはメインフレームコンピュータでも差し支えないものと理解されたい。

ＣＰＵ１０４は、メモリ１０６からのこの発明によって構成された命令に従って、コンピュータファイルへのアクセス、それらファイルがマクロを含むか否かの判定、未知ウイルス含有ウイルスの有無の判断のためのマクロ位置特定およびマクロ走査、およびウイルス検出の場合の訂正動作のための信号の供給を行う。

図２を参照すると、この発明にしたがって構成したメモリ装置１０６の好ましい実施例をより詳細に示してある。メモリ装置１０６はオペレーティングシステム１０２、アプリケーションプログラム２０４およびマクロウイルス検出モジュール２０６を蓄積している。

オペレーティングシステム２０２はワシントン州レッドモンドのマイクロソフト社から市販されているWINDOWS 3.1などのパーソナルコンピュータ用の慣用のもので構成するのが好ましい。ワードプロセッシング、表計算、作図など多様なアプリケーションプログラムの任意のものをメモリ装置１０６に蓄積できる。例えば、メモリ装置１０６にはワードプロセッシング用アプリケーションとしてマ

イクロソフトWORDを、表計算アプリケーションとしてマイクロソフトEXCELをそれぞれ蓄積できる。アプリケーションプログラム２０４は通常アプリケーションデータファイルを作成する。例えば、WORDは通常ファイル拡張子.DOCを有するデータファイルを発生する。通常のアプリケーションプログラム２０４はユーザからの反復入力なしに逐次動作を可能にするマクロを含む。慣用のマクロには、キー押下げなど相対的に単純な動作のためのもの、ファイルを開きコピーし消去するなどの動作のためのものなど多様な命令が含まれる。マクロ命令がFORMATなど下位の命令を実行するようにオペレーティングシステム（またはDOSシェル）を呼び出すこともある。マクロの用いる命令はマクロプログラム用言語をサポートするアプリケーションプログラム２０４で通常定まってくる。例えば、WORDファイル用のマクロはWordBasicプログラミング言語を用いて書かれる。

IBM社から市販されているOS/2など多様なオペレーティングシステム２０２をこの発明に代替的に用いることもできる。また、多様なアプリケーションプログラ

ム２０４を用いることもできる。この発明の実施例の一部ではWORDアプリケーションデータファイルにWordBasicコマンドを用いるマクロウイルスの検出を記載しているが、この発明が上記のような代替のオペレーティングシステム２０２および代替のアプリケーションプログラム２０４にも適用できることは当業者には理解されよう。

マクロウイルス検出モジュール２０６は、ファイルへのアクセス、それらファイルがマクロを含むか否かの判定、ウイルス含有の有無の判定のためのマクロの走査、およびウイルス含有と判定されたマクロの処置および汚染マクロ含有ファイルの訂正のためのルーチンを含む。マクロウイルス検出モジュール２０６はオペレーティングシステム２０２およびアプリケーションプログラム２０４と連携して動作する。マクロウイルス検出モジュール２０６は通常ソフトウェアで実動化するがハードウェアまたはファームウェアでも実動化できる。マクロウイルス検出モジュール２０６は図示のとおりオペレーティングシステム２０２およびアプリケーションプログラム２０４と別になっているのが好ましいが、マクロウイ

ルス検出モジュールをオペレーティングシステム２０２またはアプリケーションプログラム２０４と一体化して同様のウイルス検出訂正動作をさせることもできる。

図３を参照すると、マクロウイルス検出モジュール２０６の好ましい実施例はマクロ位置特定および復号化モジュール３０２、マクロウイルス走査モジュール３０４、マクロ処置モジュール３０６、およびファイル訂正モジュール３１０を含む。これらに加えて、ウイルス情報モジュール３０８がマクロ中のウイルス検出およびウイルス汚染マクロの処置のための比較データを供給し、データバッファ３１２がマクロウイルス検出訂正用情報を蓄積する。データバッファ３１２をいくつかの蓄積位置を含む単一のモジュールとして図示してあるが、このデータバッファ３１２の多様な機能のために複数の個別データバッファを用いることもできる。

マクロウイルス検出モジュール２０６は対象ファイルにアクセスしてマクロ含有の有無を判定する。ファイルへのアクセスはユーザが予め設定または決定したモジュール３０２のコンフィギュレーション設定に左右される。例えば、ユーザは分析のために単一のファイルだけを対象にする場合もある。また、選ばれたアプリケーションプログラム２０４に対応するファイルなどファイル群を対象にしたり、選択したディレクトリまたは蓄積領域内のファイル全部を対象にしたりすることもできる。多様な事象でファイル分析を起動できる。例えば、ユーザはウイルス走査を起動し、分析はあるアプリケーションファイルを開いた任意の時点で起動でき、またシステム１００のブートアップｎ回ごとまたは特定時間間隔ごとに完全分析を図ることもできる。マクロ位置特定および復号化モジュール３０２は、マクロウイルスを含み得る任意のファイルにアクセスし、それらファイルへのアクセスをアプリケーションプログラムの起動前、すなわちアプリケーションデータファイルを開く前に行うように構成するのが好ましい。マクロウイルスの中には、関連アプリケーションプログラムの起動とともに動作し、したがってユーザによる走査始動前の検出を要するものがあるからである。

対象ファイルの各々はマクロウイルス検出モジュール２０６によってアクセス

され、分析用にデータバッファ３１２に蓄積される。理解を容易にするために諸モジュール３０２、３０４、３０６、３０８、３１０、３１２の特定の機能に関連して単一のファイルの分析を説明するが、この発明によりいくつかのファイルを同時並行的にまたは逐次的に分析することもできる。

マクロ位置特定および復号化モジュール３０２は、対象ファイルをそれらファイルがマクロ含有型のものか否かの判定および埋め込みマクロ含有の有無の判定のために調べ、対象ファイル内でマクロを位置特定し復号化する。

このマクロ位置特定および復号化モジュール３０２はデータバッファ３１２と信号授受し、対象ファイルに分析のためのアクセスする。マクロはテンプレートファイルにも見出され、アプリケーションデータファイルに埋め込まれる。マクロ位置特定および復号化モジュール３０２は対象ファイルがテンプレートファイルであるか否かをまず判定する。この判定は拡張子をチェックすることによって行う。例えば、そのファイルがWORDアプリケーションプログラム２０４関連であればファイルを拡張子.DOTについてチェックする。この.DOT拡張子はファイルがテンプレートファイルであることを示す。

対象ファイルがテンプレートファイルであると判定されない場合は、埋込みマクロを含んでいる場合がある。例えば、.DOC拡張子つきのWORDファイルなどのアプリケーションデータファイルは埋込みマクロを含んでいることがあり得る。マクロ位置特定および復号化モジュール３０２はデータバッファ３１２に蓄積されている対象ファイルにアクセスし、そのフォーマティングが埋込みマクロを示すか否かを判定する。フォーマティングフィールドは各アプリケーションプログラム２０４の慣用の規則に従って変動し、アプリケーションプログラム２０４の製造業者から供給される。

対象ファイルがマクロを含むか否か、そのマクロが埋め込まれているか否か、テンプレートファイルの形になっているかまたはマクロをサポート可能な他の形式のファイルになっているかが判定されるとマクロは対象ファイル中で位置特定される。マクロ位置特定および復号化モジュール３０２はオペレーティングシステム２０２と信号授受する。オペレーティングシステムはWINDOWS 3.1で提供さ

れるようなオブジェクト連結埋込み（OLEまたはOLE2）などの情報共用資源を含む。この情報共用資源はアプリケーションファイルなどファイル構造の詳細を提供し、埋め込まれているオブジェクトをファイル内で位置特定できるようにする。情報共用資源コマンドはオペレーティングシステム２０２に応じて変わるが、一般に、オブジェクトを開く、特定の流れをシークする、ファイルに対して読出し書込みを行うなどの単純なコマンドである。マクロの位置特定および復号化における情報共用資源の実働化に慣用のプログラム手法を用いることができる。マクロの位置特定ののちマクロ位置特定および復号化モジュール３０２はマクロを復号化してウイルス探索のための走査ができるようにする。オペレーティングシステム２０２の情報共用資源をコヒーレントな情報へのマクロの復号化に用い、ASCII変換を走査に適した形式への復号化ずみマクロの変換に用いる。復号化ずみマクロはデータバッファ３１２に蓄積する。また、復号化ずみマクロをそのマクロの抽出元の対象ファイルに関連づける情報をデータバッファ３１２に蓄積する。

マクロウイルス走査モジュール３０４はマクロ位置特定および復号化モジュール３０２およびデータバッファ３１２と信号授受しており、したがって、モジュール３０２は復号化ずみマクロをマクロ走査モジュール３０４に供給する。このモジュール３０２に用いられるマクロ位置特定および復号化の好ましい方法は図５を参照してさらに詳細に説明する。

マクロウイルス走査モジュール３０４は、復号化ずみマクロとウイルス情報モジュール３０８からのデータとの比較に基づき既知のウイルスおよび未知のウイルスの検出のために復号化ずみマクロを走査するルーチンを含む。マクロウイルス走査モジュール３０４はマクロウイルス検出の諸態様を提供するように構成できる。例えば、走査周期を短くできるようにウイルスの特定の群だけ、すなわちウイルスのうち既知の形式のものだけ、未知の形式のものだけまたは両形式のものとも検出するように構成でき、ウイルスの最初の検出に応答して警報を発するとか、ウイルス検出の表示の前にいくつかの対象ファイルの走査を完了させておくなどの構成も可能である。

マクロウイルス走査モジュール３０４はデータバッファ３１２中の復号化ずみマクロにアクセスし、既知のウイルスについてその復号化ずみマクロを走査し、そのマクロが見出されない場合は未知のウイルスについてその復号化ずみマクロを走査する。既知のウイルスについて走査する際には、マクロウイルス走査モジュール３０４はシグネチャ走査手法を用いる。すなわちウイルス走査モジュール３０４はウイルス情報モジュール３０８と信号授受している。ウイルス情報モジュール３０８は既知のウイルスを検出する情報を含む。例えば、ウイルス情報モジュールは既知のウイルスを特定するデータまたはシグネチャのストリングを含む。ウイルス情報モジュール３０４はデータバッファ３１２の中の復号化ずみマクロにアクセスしその復号化ずみマクロを走査してウイルスシグネチャ含有の有無を判定する。この走査を行うのに状態マシーンまたは同様の手法を用いることができる。既知のウイルスシグネチャがその復号化ずみマクロ中に見出された場合は、マクロウイルス走査モジュール３０４が既知ウイルスに従ってその復号化ずみマクロを汚染マクロと特定し、その復号化ずみマクロをデータバッファ中の既知のウイルスに関連づける情報を蓄積し、マクロ処置モジュール３０６など他のモジュールがその汚染マクロを処置できるようにする。

既知のウイルスが検出されなかった場合は、マクロ走査モジュールが復号化ずみマクロの中の未知のウイルスについて走査を行う。上述のとおり、アプリケーションプログラム２０４は種々の動作にマクロが用いる命令を含むWordBasicなどのプログラム言語を含むことが多い。マクロウイルスは不要で有害な動作を行う種々の動作および命令を用いる。

通常のアプリケーションプログラム２０４はマクロをテンプレートファイルに提供することによってマクロをサポートする。テンプレートファイルはワードプロセシング設定など飾りほかの設定を含む。また、テンプレートファイルはマクロを含み得る。通常は、グローバルテンプレートファイルはデータファイルのために設定およびマクロを提供する。例えば、マイクロソフトWORDについては、グローバル設定およびマクロのプールはテンプレートファイルNORMAL.DOTに常駐する。アプリケーションプログラム２０４がデータファイルを開くと、それによっ

てグローバルテンプレートファイルがまず開き、グローバル設定およびマクロをロードし、そのあとでデータファイルを開く。通常のデータファイルは埋込みマクロを含まないアプリケーションプログラム２０４を表すようにフォーマットされている。しかし、データファイルはテンプレートファイルを含まない旨をアプリケーションプログラム２０４に表示するようにフォーマットすることもできる。

ある種のマクロウイルスは汚染された文書ファイルをテンプレートフォーマットで保存し文書ファイルまたはデータファイル拡張子（.DOC）は消去しないまま保存するようにする。したがって、汚染マクロは外見上アプリケーションデータファイルの文書の中に埋め込まれ得る。その種のマクロには「AutoOpen」、「AutoExec」、「AutoClose」など、すなわちデータファイルを開いた際にマクロを実行させるようマクロが含まれる。したがって、ユーザは通常のデータファイルに見えるものを開こうとすることはできるが、そうすると、埋込みマクロを自動的に実行させることになる。マクロウイルスも他のファイルで複製を生じる。例えば、マクロウイルスはデータファイル中に自分自身をコピーしてそのデータファイルへの通常のファイル拡張子を維持しながらテンプレートフォーマットを表示するようにデータファイルをフォーマットすることがよくある。

マクロウイルスに汚染されたファイルはそのフォーマットを変えられてしまったり、汚染されたデータファイルが更新ずみのフォーマット情報とともにマクロウイルスによって保存されたりする。また、汚染されたマクロがグローバルテンプレートにコピーされ、その結果、他のファイルが開かれる際にそれら他のファイルに広がり得る。

マクロウイルス走査モジュール３０４はマクロウイルスに使われる可能性の高いマクロ命令組合せ、換言すれば容疑命令組合せを検出するルーチンを含む。マクロウイルス走査モジュール３０４の検出する容疑命令の一つの組合せはマクロイネーブル化命令およびマクロ複製命令である。マクロイネーブル化命令はファイルのフォーマッティングが実行用マクロ含有ファイルを表示するように設定可能な命令である。例えば、ファイルフォーマッティングの設定を、ファイルを開

いた際にテンプレートファイルがアプリケーションプログラム２０４で実行されるようにテンプレートファイルを表示するように行うことができる。マクロ複製命令はマクロウイルスの複製を可能にする命令である。マクロイネーブル化命令とマクロ複製命令との組合せはマクロウイルスを表示する。すなわち、そのような命令は先行ファイルにおけるマクロの複製および実行を可能にするからであり、これらはマクロウイルスの二つの通常の特徴を構成する。

容疑命令の組合せを特定するためにマクロウイルス走査モジュール３０４はウイルス情報モジュール３０８からの比較データにアクセスする。この比較データは、復号化ずみマクロの中の容疑命令の組合せの特定のための命令識別子組を含む。それら命令識別子組の好例は第１および第２の容疑命令識別子を含む。命令識別子は二進のストリングであり、復号化ずみマクロの走査はこれら二進のストリング、すなわち容疑命令をそれらマクロが含んでいるか否かを判定するように行う。命令識別子の組で定義した容疑命令の組合せをマクロが含むと判定された場合は、そのマクロがそのデータ組に対応する未知のウイルスで汚染されていると判定する。マクロウイルス走査モジュール３０４は復号化ずみマクロを未知ウイルスによる汚染マクロとフラグ表示し、その復号化ずみマクロをデータバッファ３１２中における未知ウイルス検出に導いた命令識別子に関連づける情報を蓄積し、マクロ処置モジュール３０６など他のモジュールが汚染マクロを然るべく処置できるようにする。マクロウイルス走査モジュール３０４の検出した容疑命令の組合せ、すなわちマクロウイルスイネーブル化およびマクロウイルス複製の命令などの命令の組合せは図６を参照してさらに詳述する。

ウイルス情報モジュール３０８はマクロウイルス検出モジュール２０６中の他のモジュール３０２、３０４、３０６、３１０、３１２から分離するのが好ましい。それによって、マクロウイルス検出用の情報の更新が容易になる。例えば、ウイルス情報３０８はフロッピーディスクなどの媒体から得た新しい情報をコピーすることによって更新できる。また、コンピュータシステム１００の通信ユニット１１２経由、またはネットワークリンク（図示してない）経由でアクセスしたインターネット資源から新たな情報をダウンロードすることもできる。分離し

たウイルス情報モジュール３０８によって情報転送がより容易になり、更新も容易になり、したがって未知ウイルスを含むウイルスに対するシステム１００の保護が強化される。

マクロ処置モジュール３０６はデータバッファ３１２およびマクロウイルス走査モジュール３０４と信号授受し、したがって対象ファイルからの復号化ずみマクロの中のウイルスの検出に関する情報を受ける。マクロ処理モジュール３０６は、マクロウイルス走査モジュール３０４が復号化ずみマクロの中に既知または未知のウイルスを検出したか否かを判定するための復号化ずみマクロの状態のチェックのルーチン、復号化ずみマクロからマクロウイルスを除去するルーチン、および処置ずみマクロの完全無欠性を検証するルーチンを含む。

マクロ処置モジュール３０６はデータバッファ３１２の中の復号化ずみマクロにアクセスして復号化ずみマクロの状態をチェックし、マクロウイルス走査モジュール３０４が既知のウイルスを検出したか否かを判定する。そのマクロの状態はデータバッファ３１２内では状態フラグなどの情報で表される。また、データバッファ３１２は復号化ずみマクロをそのマクロに含まれる既知のウイルスと関連づける情報を蓄積している。上述のとおり、この情報はマクロウイルス走査モジュール３０４から供給される。マクロウイルス走査モジュール３０４に適切な情報を蓄積しマクロ処置モジュール３０６と直接に信号授受することもできる。既知ウイルス判定フラグが表示されている場合は、マクロ処置モジュールは復号化ずみマクロからその既知マクロを除去するための既知ウイルス関連情報を用いる。既知のマクロウイルスを復号化ずみマクロから選択的に除去し非汚染命令と置換して、そのマクロの残余の部分がそれ以降の動作のために保持するのが好ましい。処置ずみマクロはマクロ処置モジュール３０６によってデータバッファ３１２内に区別して蓄積される。次に、処置ずみマクロの完全無欠性をマクロ処置モジュールで確認し、完全無欠性が維持されている場合はその処置ずみマクロに有効とフラグ表示する。処置ずみマクロの完全無欠性が維持されていなかった場合は、処置ずみマクロを無効と表示する。マクロの完全無欠性のチェックは、残余の命令が無傷であるか否かの確認および命令の逐次連結が無傷の状態に留まっているか否かの確認によって行う。マクロの完全無欠性の検証は、ファイル訂正

モジュール３１０など他のモジュールが、汚染マクロを処置ずみマクロで置換するまたは対象ファイルから汚染マクロ消去するに留めるなどの代替処置を決定することを可能にする。

マクロウイルス走査モジュール３０４が未知のウイルスを検出した場合は、その未知のウイルスの影響を除去するようにマクロ処置モジュール３０６がそのマクロを処置する。既知ウイルス処置プロトコルと同様に、マクロ処置モジュール３０６はデータバッファ３１２と信号授受して未知のウイルスの検出の有無を判定し、有りと判定した場合はウイルス有りの判定に導いた命令識別子を特定する。マクロの中の未知ウイルスの検出に導いた容疑命令識別子の組をそのマクロの訂正に用いる。各命令識別子は、各命令の識別およびマクロからの除去を可能にするように一つ以上の容疑命令と関連づけてある。ウイルス訂正用のマクロ処置モジュール３０６はマクロを復号化して訂正に備え、またはマクロウイルス走査モジュール３０４で復号化ずみのマクロにアクセスする。ここでも容疑命令を復号化マクロから除去し非汚染命令と置換するのが好ましい。処置ずみマクロはデータバッファ３１２に蓄積し、ファイル訂正モジュール３１０など他のモジュールによるアクセスに備える。上記の対既知ウイルス処置プロトコルと同様に、処置ずみマクロの完全無欠性を検証し、そのマクロに然るべきフラグ表示を立てる。好適なマクロ処置ルーチンは図７を参照してより詳細に述べる。

ファイル訂正モジュール３１０はデータバッファ３１２およびマクロ処置モジュール３０６ほかのモジュールと信号授受し、対象ファイルからのマクロで検出されたウイルスに関する情報をその信号授受を通じて受ける。ファイル訂正モジュール３１０は汚染ファイルの表示があった場合の処置動作のためのルーチンを含む。マクロ処置モジュール３０６の中のルーチンは自動的にまたはユーザ許可時のみに種々の処置動作を行うように構成できる。例えば、ファイル訂正モジュール３１０は汚染マクロ含有の対象ファイルをコピーし、その汚染マクロ処置ずみマクロと置換し、ユーザへの通知なしに対象ファイルを訂正ずみと置換することができる。ファイル訂正モジュール３１０も、訂正動作の諸段階でユーザに進行の可否を問い合わせるプロンプトを生ずるように構成できる。この動作がコン

ピュータシステム１００の入力装置１０８および表示装置１０２を用いて対話式に行われることはもちろんである。例えば、ファイル訂正モジュール３１０が対象ファイルからのマクロの中にある種のウイルスまたは未知のウイルスが検出されたことをユーザに表示する。次に、ユーザは対象ファイルを訂正ずみファイルに置換することを望むか否かの問合せを受ける。ファイル訂正モジュール３１０の構成の仕方、および諸段階におけるプロント表示の仕方が多様であることは当業者に理解されよう。

ファイル訂正モジュール３１０はデータバッファ３１２と信号授受し、それによって汚染マクロ含有の対象ファイルを表示する。ファイル訂正モジュール３１０は汚染マクロ含有の対象ファイルにアクセスし、その対象ファイルのコピーをそのファイルの訂正のためにデータバッファ３１２に蓄積する。マクロ位置特定および復号化モジュール３０２、マクロウイルス走査モジュール３０４およびマクロ処置モジュール３０６について上述したとおり、データバッファ３１２は対象ファイルと処置ずみマクロとの関係やその処置ずみマクロの有効無効などに関する情報および検出ウイルスの種類に関する情報を蓄積している。ファイル訂正モジュール３１０はデータバッファ３１２の中のマクロ有効性フラグをチェックして、マクロ処置モジュール３０６による検出ウイルスの除去および処置ずみマクロの完全無欠性の維持が可能であったか否かを判定する。処置ずみマクロの完全無欠性が維持されていなかった場合は、ファイル訂正モジュール３１０がデータバッファ３１２の中の対象ファイルをその中の汚染マクロを処置ずみマクロと置換することによって訂正する。ウイルス汚染マクロを汚染ファイルの中でまず位置特定する。この動作はマクロ位置特定および復号化モジュール３０２との信号授受によって行うことができ、またモジュール３０２と同様にオペレーティングシステム２０２の情報共用資源にマクロ位置特定のためにアクセスできるファイル訂正モジュール３１０によって独立に行うこともできる。汚染を受けた対象ファイルのコピーをデータバッファ３１２に蓄積する。次に処置ずみマクロをそのマクロ抜きの対象ファイルのバージョンに加えて訂正ずみファイルを生ずる。この訂正ずみファイルを元の位置の対象ファイルへの置換に用いる。これは対象

ファイルを訂正ずみファイルで直接に置換する手法である。代替的に、対象ファイルを消去または上書きし、訂正ずみファイルを別の位置に蓄積することもできる。

無効とフラグ表示された対象ファイルに対応する処置ずみマクロは汚染マクロとの置換には用いないのが好ましい。その場合は、ファイル訂正モジュール３１０によって種々の代替的訂正動作を行うことができる。例えば、対象ファイルがウイルスを含んでいることをユーザに知らせ、汚染マクロを置換なしに汚染ファイルから除去し、または対象ファイルを除去することができる。

図４を参照すると、マクロの中の未知のウイルスを検出する方法４００流れ図が示してある。ウイルス検出のための対象ファイルにまずアクセスする。対象ファイルは通常メモリ１０６内にあり、それら対象ファイルをデータバッファ３１２にコピーしてマクロウイルス検出モジュール２０６がそれら対象ファイルからマクロウイルスを検出し除去できるようにする。対象のファイル一つだけについて処理を詳述するが、この発明では多数のファイルにアクセスして検査することが可能である。ファイルへのアクセスのタイミングと範囲は、マクロウイルス検出モジュール２０６の説明において上述したとおり、このモジュール２０６がどのように構成されているかに左右される。すなわち、多様なファイルを対象にでき、ユーザに選択可能な条件に基づいてウイルス検出を起動できる。しかし、マクロウイルス検出モジュール２０６はアプリケーションプログラム２０４の起動の必要なしにアプリケーションプログラム２０４起動で動作するマクロウイルスの検出および除去が可能になるようにするのが望ましい。

マクロ位置特定および復号化モジュール３０２は対象ファイルにアクセスしデータバッファに供給したのち、その対象ファイル中のマクロの有無を判定し、それらマクロの位置特定および復号化に進む。ファイルの位置特定および復号化の好ましい方法５００は図５を参照してより詳細に述べる。次に、マクロが位置特定および復号化モジュール３０２によって見出されたか否かを判定する（４４０）。マクロが対象ファイルの中にあると判定された場合（４４０）は、マクロをマクロウイルス走査モジュール３０４で走査し（６００）、そのマクロがマクロ

ウイルスで汚染されているかどうかを判定する。対象ファイルの中にマクロがないと判定された場合（４４０）は、このマクロウイルス検出方法は終了する。好ましい走査の方法（６００）は図６を参照してより詳細に述べる。走査においてウイルスが検出された場合（４６０）は、汚染マクロをマクロ処置モジュール３０６で処置する（７００）。走査においてウイルスが検出されなかった場合（４６０）は、このウイルス検出方法は終了する。マクロ処置の好ましい方法は図７を参照してより詳細に述べる。マクロ処置ののち（７００）、汚染した対象ファイルに

対して訂正動作（８００）を行い、そのあとでマクロウイルス検出の好ましい方法を終了する。好ましい訂正方法は図８を参照して詳述する。

図５を参照すると、この発明によるマクロ位置特定および復号化の好ましい方法５００が示してある。アプリケーションデータファイルやテンプレートファイルなど種々のファイルを対象にすることができる。マクロ位置特定および複合化モジュール３０２は、対象ファイルのマクロ含有の有無の判定をそのファイルの所属種類の判定（ステップ５０５）によってまず行うルーチンを含む。この判定は対象ファイルのファイル拡張子をチェックすることによって行う。所属ファイル種類のチェックによって、対象ファイルがテンプレートファイルであるか否かを判定できる。そのファイルがテンプレートファイルである場合は、マクロ位置特定および復号化モジュール３０２は対象ファイルにおける埋込みマクロの有無を判定する必要はない。したがって、そのファイルがテンプレートファイルであるとステップ５１０で判定された場合は、そのファイルの中のどのマクロも位置特定され（５２５）復号化され（５３０）、マクロウイルス走査モジュール３０４によるウイルスについての走査に備える。復号化ずみのマクロはいずれもデータバッファ３１２に蓄積され（５３５）、マクロウイルス走査モジュール３０４によるアクセスに備える。

そのファイルがテンプレートファイルまたはそれ以外のマクロを含有し得る種類のファイルでないとステップ５１０で判定された場合は、ステップ５１５で対象ファイルを調べて埋込みマクロの有無を判定する。埋込みマクロがない場合は

、対象ファイルがマクロを含まないとステップ５４０で判定して、この好ましい方法は終了する。ファイルが埋込みマクロを含むか否かの判定はファイルフォーマッティングをチェックすることによって行う。例えば、一つのフォーマットは、アプリケーションデータファイルが拡張子の表示如何に関わらずテンプレートファイルを含む旨をアプリケーションプログラム２０４に示すことを可能にする。そのファイルフォーマットがテンプレートファイル含有を表す場合は、その対象ファイルは埋込みマクロを含むかもしれない。ステップ５２０においてファイルが埋込みマクロを含まないと判定された場合は、ステップ５４０においてマクロは対象ファイル中にないと判定し、この好ましい方法５００は終了する。

対象ファイルは埋込みマクロを含まないとステップ５２０で判定された場合、

または対象ファイルはマクロ含有テンプレートファイルであるとステップ５１０で判定された場合は、そのマクロをステップ５２５で位置特定しステップ５３０で復号化する。位置特定および復号化モジュール３０２は、WINDOWS 3.1オペレーティングシステムで提供されているようなオブジェクト連結埋込み(OLEまたはOLE2) などのオペレーティングシステム２０２情報共用資源を用いるルーチンを含む。位置特定および復号化モジュール３０２について上述したとおり、情報共用資源は、ファイルに一体化されたオブジェクトの位置特定および復号化を可能にするように、ファイル構造に関する詳細、すなわちアプリケーションファイルやテンプレートファイルの詳細を提供する命令を含む。慣用のプログラム作成手法をマクロ位置特定および復号化のための情報共用資源の実働化に使うことができる。マクロを位置特定し復号化したのち二進符号に変換し（例えばASCII変換により）、ステップ５３５においてデータバッファ３１２に蓄積してウイルスについての走査ができるようにする。マクロ位置特定および復号化モジュール３０２は、復号化マクロの蓄積のほかに、復号化ずみマクロと対象ファイルとの間の関連を維持し蓄積し、マクロウイルス走査モジュール３０４、マクロ処置モジュール３０６およびファイル訂正モジュール３１０が対象ファイルの正しい走査および処置、および訂正をできるようにする。ステップ５３５でマクロの復号化および蓄積が終了したのち、この好ましい位置特定および復号化の方法５００は終

了する。

図６の流れ図を参照すると、この発明によるマクロ中ウイルス検出の好ましい方法６００が示してある。マクロウイルス走査モジュール３０４は、位置特定および復号化モジュール３０２の提供する復号化ずみマクロ情報にアクセスし、復号化ずみマクロの情報とウイルス情報３０８との比較によりウイルスの存在を検出するルーチンを含む。

第１のステップ６０５において、復号化マクロを既知のウイルスについて走査する。既知のウイルスについて走査するために、マクロウイルス走査モジュール３０４はシグネチャ走査手法を用いる。マクロウイルス走査モジュール３０４はデータバッファ３１２中の復号化ずみマクロにアクセスし、ウイルス情報モジュールの提供するウイルスシグネチャをそれらマクロが含むか否かを判定する。ステップ６１０において、既知ウイルス走査ステップ６０５に基づく既知ウイルスを復号化ずみマクロが含むか否かを判定し、既知ウイルス有りの場合は、ステップ６１５においてマクロウイルス走査モジュールがそのマクロを既知ウイルスによる汚染有りとフラグ表示しその復号化ずみマクロとその既知マクロとを関連づける情報をデータバッファ３１２に蓄積する。

走査ステップ６０５で既知ウイルスが検出されなかったとステップ６１０で判定した場合は、マクロウイルス走査モジュール３０４が復号化ずみマクロの中の未知のウイルスについて走査を行う。ステップ６１５において、マクロウイルス走査モジュール３０４は未知のウイルスの検出のための一連の命令識別子を取り入れる。マクロ走査モジュール３０４はマクロウイルスに使われそうな命令を検出する。これら命令は容疑命令とも呼ぶ。容疑命令の特定の組合せはマクロウイルスに使われる可能性が高い。容疑命令の組合せの探索によって、誤ったウイルス検出を回避できる。二つ（またはそれ以上）の互いに異なる容疑命令を含むマクロは汚染されている可能性がごく高いからである。

マクロウイルス走査モジュール３０４の説明で述べたとり、通常のアプリケーションプログラム２０４はテンプレートファイルにマクロを提供する。通常はアプリケーションデータファイルはグローバルテンプレートファイルを使うが、埋

込みマクロを含むことを示すようにフォーマットすることもできる。例えば、WORDファイルをテンプレートファイル含有の表示のために.DOTフォーマットで保存することができる。多くのマクロウイルスが、汚染ドキュメントファイルをテンプレートフォーマット（.DOT）で保存させ、ドキュメントまたはデータファイル拡張子（.DOC）を不変のまま保持させる。したがって、汚染マクロは外見上単なるアプリケーションデータファイルに見えるドキュメントに埋め込まれるかもしれない。マクロウイルスは他のファイルに自分自身の複製を作る。例えば、マクロウイルスはデータファイルに自分自身をコピーして、データファイル用の通常のファイル拡張子を維持しながらデータファイルをテンプレートフォーマット表示状態にフォーマットすることが多い。

マクロ走査モジュール３０４の検出した容疑命令の一つの組合せはマクロイネーブル化命令およびマクロ複製命令である。マクロネーブル化命令は、そのファイルが実行用のマクロを含むことを表すようにファイルをフォーマットするものである。例えば、ファイルフォーマッティングをテンプレートファイルを表すように設定して、アプリケーションプログラム２０４がテンプレートファイルをフ

ァイルが開いたとき実行するようにすることができる。マクロ複製命令はマクロウイルスの複製を可能にする命令である。マクロイネーブル化命令とマクロ複製命令との組合せはマクロウイルスを表す。それら命令によって先行ファイルにおけるマクロウイルスの二つの通常の特徴、すなわちマクロの複製および実行が可能になるからである。

マイクロソフトWORDファイルなど特定のアプリケーションファイルでは、ファイルフォーマットフィールド.formatを１に設定すると、アプリケーションプログラム２０４（WORD）がファイルに埋込みマクロが含まれているものと判断し、適当な始動により、ファイルに埋込みずみの任意のマクロにアクセスしてそれを実行する。すなわち、ファイル中で.formatを１に設定することによって、そのファイルの中のマクロの実行をイネーブルし、先行ファイルでそのような設定の提供を求める命令はすべてマクロイネーブル化命令とみなす。例えば、命令「if dlg.format=0, then dlg.format=1」は先行ファイルで.formatを０から１に変

えることを可能にするのでマクロイネーブル化命令である。命令「FileSaveAsa$,1」など他の命令は元のファイルを保持し、ファイルが埋込みマクロを含み得る旨を表示するフォーマットなど別のフォーマットでそのファイルの追加のコピーを保存する。したがって、この種の命令もマクロイネーブル化命令である。ファイル中でマクロウイルス実行をイネーブルする種々の代替的命令が認識されよう。

マクロウイルス複製命令はマクロウイルスの反復を可能にする種類のものである。例えば、命令「MacroCopy」はマクロをコピーし、そのマクロが汚染している場合は有害な命令全部、すなわち送信元から宛先まで全部をコピーする。命令「Organizer.copy」など上記以外の命令もマクロウイルス複製を容易にする。種々の代替的命令がマクロウイルス複製を容易にできることを理解されたい。

マクロ位置特定および復号化の好ましい方法５００について述べたとおり、対象ファイルからのマクロ命令は位置特定されたのち分析のために２進符号に変換される。特有の２進符号は容疑命令にも対応する。例えば、マクロウイルスイネーブル化命令「ifdlg.format=0thendlg.format=1」はマクロウイルス複製命令「MacroCopy」と同様に特定の対応２進符号を有する。したがって、ウイルス情報モジュール３０８から得られる比較データ（６１５）は第１および第２の命令用の２進符号または２進符号の特有部分をそれぞれ含み、それによって対象ファイルからのマクロの中の第１および第２の命令を特定する。

また、この発明によって特有の２進符号部分がいくつかの容疑命令に対応すると判定された。例えば、２進ストリング「73 CB 00 0C 6C 01 00」（１６進表記）はいくつかのマクロウイルスイネーブル化命令中に見出される命令部分「.format=1」に対応する。また、例えば、上記命令「ifdlg.format=0thendlg.format=1」、「ifbewaardlg.format=0thenbewoordlg.format=1;およびFileSaveAs.Format=1」、および「FileSaveAs.Name=Filename$(),.Format=1」は２進ストリング「73 CB 00 0C 6C 01 00」は２進ストリングを含む。したがって、この発明はこの73 CB 00 0C 6C 01 00など特定のストリングを複数の互いに異なる容疑マクロ命令の検出のための識別子として用いる。

ウイルス情報モジュール３０８の中の比較データにはいくつかの組の命令識別子を含めるのが好ましい。容疑命令の種々の組合せをそれら命令識別子組の利用により検出できる。種々のマクロウイルスイネーブル化命令やマクロウイルス複製命令をこれら命令識別子の各組を用いて識別できる。命令識別子はマクロウイルスイネーブル化命令およびマクロウイルス複製命令に限られない。例えば、コンピュータハードディスク装置に認証および命令なしに再初期化を行わせる命令、すなわちユーザへの通知なしの再初期化を可能にするようにシステム設定を変更する命令は容疑命令組合せとして使用可能である。

図９を参照すると、ウイルス情報モジュール３０８に蓄積した命令識別子の好例を含むデータ表が示してある。この好例データ表９００はいくつかの互いに異なる命令識別子に対応する行９０２を含む。また、命令識別子９０３の組を識別する列、命令識別子ＩＤ番号９０４および命令識別子２進符号のテキストおよび対応１６進表記９０５もこの表に含めてある。命令識別子の各組に二つの命令識別子を含めるのが好ましいが、付加的な命令識別子は一つの組に含め得る。また、マクロウイルス判定は三つの命令識別子のうちの二つまたはそれ以外の識別子小群の検出に基づいて行うことができる。データ表９００は例示にすぎない。ウイルス情報モジュール３０８への比較データの蓄積は種々の手法で行うことができる。

図６の流れ図を参照すると、ステップ６１５においてマクロウイルス走査モジュール３０４で一組の命令識別子を得たのち復号化ずみマクロを走査して、命令識別子の識別したような容疑命令の組合せを含むか否かを判定する。ステップ６２０において復号化ずみマクロを第１の命令識別子を用いて走査する。例えば、復号化ずみマクロを走査して（６２０）、命令識別子９００の第１の組の中の第１の命令識別子に対応するストリング73 CB 00 0C 6C 01 00が存在するか否かを判定する。ステップ６２０における走査は状態マシーン、すなわち復号化ずみマクロを走査して上記ストリングの有無を判定する状態マシーンで行う。ステップ６２５において、第１の容疑命令識別子が復号化ずみマクロに存在するか否かを判定する。この第１の命令識別子に対応する命令はないと判定した場合は（６２

５）、ステップ６４５でこの命令識別子組にしたがってそのマクロを非汚染と判定し、このマクロウイルス走査方法６００を終了する。

ステップ６２５において上記第１の命令識別子があると判定した場合は、ステップ６３０で復号化ずみマクロを走査し、第２の命令識別子の有無を判定する。ステップ６３５でこのマクロが第２の容疑命令識別子を含むと判定した場合は、ステップ６４０で復号化ずみマクロをその命令識別子組対応の未知のウイルスによる汚染マクロとフラグ表示する。未知ウイルス検出に導いた命令識別子組に復号化ずみマクロ関連づける情報をデータバッファ３１２に蓄積して、マクロ処置モジュール３０６など他のモジュールが汚染マクロを然るべく処置できるようにする。

ステップ６３５で第２の容疑命令識別子なしと判定した場合は、ステップ６４５でマクロウイルス走査モジュール３０４が命令識別子組にしたがって復号化マクロ中には未知ウイルスなしと判定し、マクロウイルス走査方法６００を終了する。ステップ６３５におけるこの判定は命令識別子の単一の組について行われる。命令識別子の上記以外の組は復号化マクロを反復的に比較して、未知ウイルスの判定を可能にする。また、共通の第１の命令識別子の有無は、種々の代替的な第２の命令識別子の探索の前に判定できる。

図７の流れ図を参照すると、好適な汚染マクロ処置方法７００が示してある。ステップ７０５において、未知ウイルス判定フラグをチェックしてマクロウイルス走査モジュール３０４が復号化ずみマクロの中に既知のウイルスを検出したか否かを判定する。既知ウイルス判定フラグはデータバッファ３１２の中のマクロ処置モジュール３０６、すなわち既知ウイルスの検出に用いたウイルス情報に復

号化ずみマクロを関連づけるモジュール３０６に供給する。ステップ７１５でこのウイルス情報を用いて既知ウイルスを復号化ずみマクロから除去する。マクロからのウイルスの除去は、ウイルスを非汚染命令（no-opなど）に置換することによって行う。ウイルスは既知であるから、マクロの正常部分がそのまま残るように選択的に除去できる。ウイルス除去ののち、ステップ７３５において処置ずみマクロをチェックしてその完全無欠性を検証する。処置ずみマクロの完全無欠

性が維持されていると判定した場合は、ステップ７４５で処置ずみマクロにマクロ処置モジュール３０６で有効とフラグ表示する。処置ずみマクロは、そのマクロの有効性関連データとともにデータバッファ３１２に蓄積しその状態に保つ。ステップ７４０において上記完全無欠性が維持されていないと判定された場合は、ステップ７５０においてその処置ずみマクロを無効とフラグ表示し、関連情報をデータバッファ３１２に同様に蓄積する。

図７に戻って、既知のウイルスは復号化ずみマクロにないとマクロ処置モジュール３０６が判定した場合は、未知のウイルスを選択的に除去するようにマクロを処置する。復号化ずみマクロの中の未知のウイルスの検出に用いた命令識別子の組をデータバッファ３１２の中のマクロ処置モジュール３０６に利用できる。この命令識別子の組は例えば第１および第２の容疑命令識別子を含む。ステップ７２０では第１の容疑命令識別子をその識別子対応の容疑命令の各々の位置特定に用いる。マクロウイルス走査モジュール３０４の用いる命令の検出に関連づけて説明した手法を用いて上記命令を位置特定するのに復号化ずみマクロを走査できる。命令識別子が命令全体でなく命令の断片と対応する場合は、マクロ処置モジュール３０６が検出された断片の各々を命令全体に関連づける。この関連づけはマクロの用いるプログラム用言語に左右される。この関連づけには慣用の手法を使用できる。ステップ７２５では追加の容疑命令識別子を対応の容疑命令の検出に用いる。次に、ステップ７３０で、位置特定した容疑命令を置換する。既知のウイルスストリングの置換と同様に、容疑命令を非汚染命令と置換するのが好ましい。マクロの完全無欠性を検証し、完全無欠性維持の有無に応じて処置ずみマクロにフラグ表示し、マクロ処置方法７００を終了する。

図８を参照すると、この発明による好ましい訂正方法８００が示してある。ファイル訂正モジュール３１０はデータバッファ３１２および種々のモジュール３０２、３０４、３０６、３０８、３１０と信号授受し、検出されたマクロウイルスや汚染マクロ含有の検出された対象ファイルなどの情報にアクセスする。ステップ８０５でマクロウイルス含有の対象ファイルをデータバッファ３１２に蓄積する。対象ファイルを元の位置でアクセスしその汚染マクロとともにデータバッ

ファ３１２にコピーするのが好ましい。次に、ファイル訂正モジュール３１０は対象ファイルの中のマクロを処置ずみマクロに置換するか代替的訂正手法を用いるかによって訂正動作を行う。ステップ８１０でマクロ有効性フラグをチェックし、対象ファイル対応の処置ずみマクロの完全無欠性を判定する。処置ずみマクロが有効と表示されている場合は、ファイル訂正モジュール３１０が対象ファイルの中の汚染マクロを処置ずみマクロで置換する。ステップ８１０では、汚染マクロは対象ファイル中で位置特定される。この動作はオペレーティングシステム２０２の情報共用資源（OLE）を用いて行う。ステップ８２０では、位置特定したマクロを情報共用資源の利用により対象ファイルから除去し、そのマクロ抜きの対象ファイルのバージョンをデータバッファ３１２に蓄積する。ステップ８２５では、マクロ処置モジュール３０６の発生した処置ずみマクロをそのマクロ抜きの対象ファイルのバージョンに加えて訂正ずみファイルを発生する。ステップ８３０では、この訂正ずみファイルを元の位置で対象ファイルの代わりに入れるのに用いる。訂正ずみファイルで初めからの対象ファイルを直接に置換することもできる。また、対応の対象ファイルは消去または上書きでき、訂正ずみのファイルは任意の位置に蓄積できる。

ステップ８１０に戻ると、対象ファイル対応の処置ずみマクロを無効とフラグ表示し、この処置ずみマクロは汚染マクロの置換には用いないのが好ましい。すなわち、ステップ８３５では、ユーザのコンフィギュレーション設定に応じて、ファイル訂正モジュールで代替の訂正動作を行う。多様な代替の訂正過程、すなわち対象ファイルがウイルスを含む旨をユーザに知らせる、対象ファイルからの汚染ファイルの除去を置換なしに行う、または目標ファイルを消去するなど代替の訂正過程が認識されよう。

特定の実施例を参照してこの発明を上に説明してきたが、種々の変形が可能であることは当業者に認識されよう。例えば、種々のモジュールについてアクセス、位置特定、復号化、検出および訂正の系列を説明してきたが、マクロ中で未知の

ウイルスを検出する際に等価の機能を発揮する通常のモジュールに種々のプロセ

スを取り込めることは理解されよう。上記実施例のこれらのおよびこれら以外の変形および改変をこの発明は提供するものであり、この発明の範囲は添付請求の範囲のみによって限定されるものである。

ブロック図（図１－３）および流れ図（図４－８）の各構成部分の対応訳

（図１）

102　表示装置　　104　中央処理装置（CPU）

106　メモリ装置　　108　入力装置

110　データ蓄積装置　　112　通信ユニット

（図２）

202　オペレーティングシステム　　204　アプリケーションプログラム

206　マクロウイルス検出モジュール

（図３）

206　マクロウイルス検出モジュール

302　マクロ位置特定および復号化モジュール

304　マクロウイルス走査モジュール

306　マクロ処置モジュール　　308　ウイルス情報モジュール

310　ファイル訂正モジュール　　312　データバッファ

（図４）

420　ファイルにアクセスする

500　ファイルからのマクロを位置特定し復号化する

440　マクロ有り？

600　ウイルスについてマクロを走査する

460　ウイルス検出された？

700　汚染マクロを処置する

800　汚染ファイルに対して訂正動作を行う

（図５）

505　ファイル種類を判定する

510　テンプレートファイル？

515　埋込みマクロ含有の有無を判定するようにファイルを調べる

520　埋込みファイル有り？

525　ファイル中でマクロを位置特定する

530　走査のためにマクロを復号化する

535　復号化ずみマクロをバッファに蓄積する

540　ファイル中にマクロ常駐なしと判定する

（図６）

605　既知ウイルスについて走査する

610　既知ウイルス有り？

615　未知ウイルス特定用の比較データを取り込む

620　比較データからの第１の命令識別子を用いマクロを走査する

625　第１の容疑命令有り？

630　比較データからの第２の命令識別子を用いマクロを走査する

635　第２の容疑命令有り？

640　この命令識別子組対応の未知ウイルスで汚染とマクロにフラグ表示する

645　この命令識別子組によるマクロの汚染なしと判定する

650　既知ウイルスによる汚染有りとマクロにフラグ表示する

（図７）

705　既知ウイルス判定フラグをチェックする

710　既知ウイルス有り？

715　既知ウイルスをマクロから除去する

720　第１命令識別子対応の各容疑命令を位置特定する

725　追加の命令識別子対応の各容疑命令を位置特定する

730　特定ずみの各容疑命令を非汚染命令で置換する

735　処置ずみマクロの完全無欠性を検証する

740　完全無欠性は維持されている？

745　処置ずみマクロを有効とフラグ表示する

750　処置ずみマクロを無効とフラグ表示する

（図８）

805　対象ファイルをデータバッファに蓄積する

810　マクロ有効性フラグ表示有り？

815　対象ファイル中でマクロを位置特定する

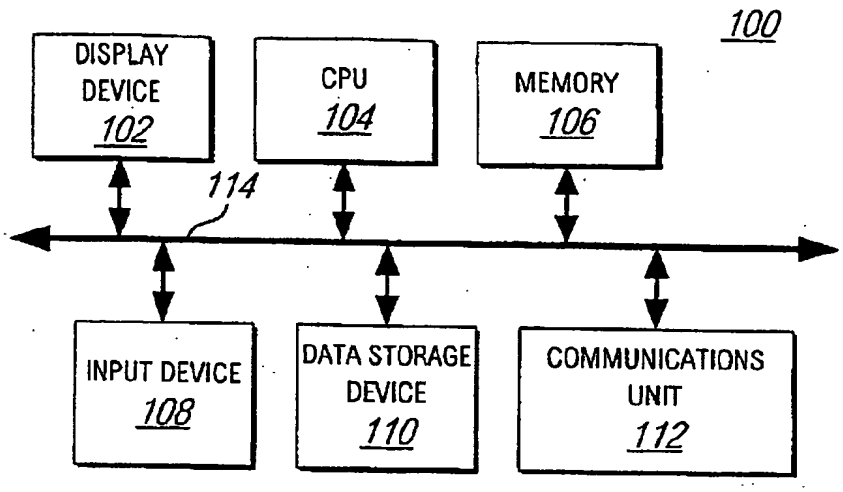
820　対象ファイルからマクロを除去しマクロなしファイルの複製を蓄積する

825　処置ずみマクロをマクロ除去ずみファイルに加える

830　対象ファイルを訂正ずみファイルで置換する

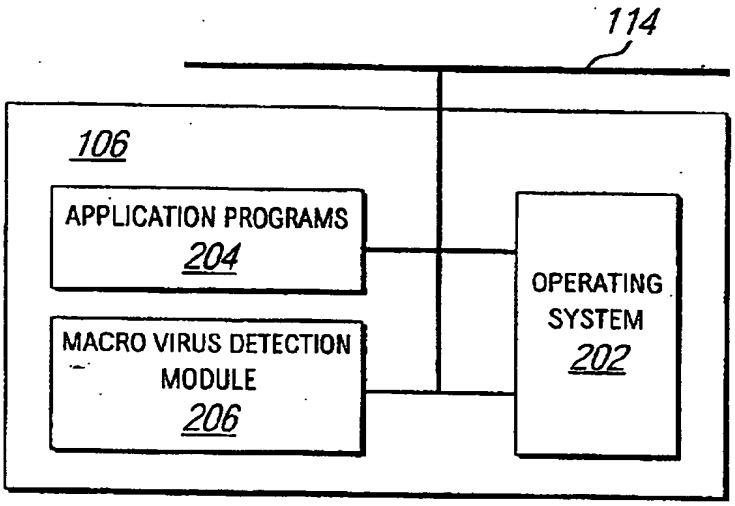
【図１】

FIG. 1



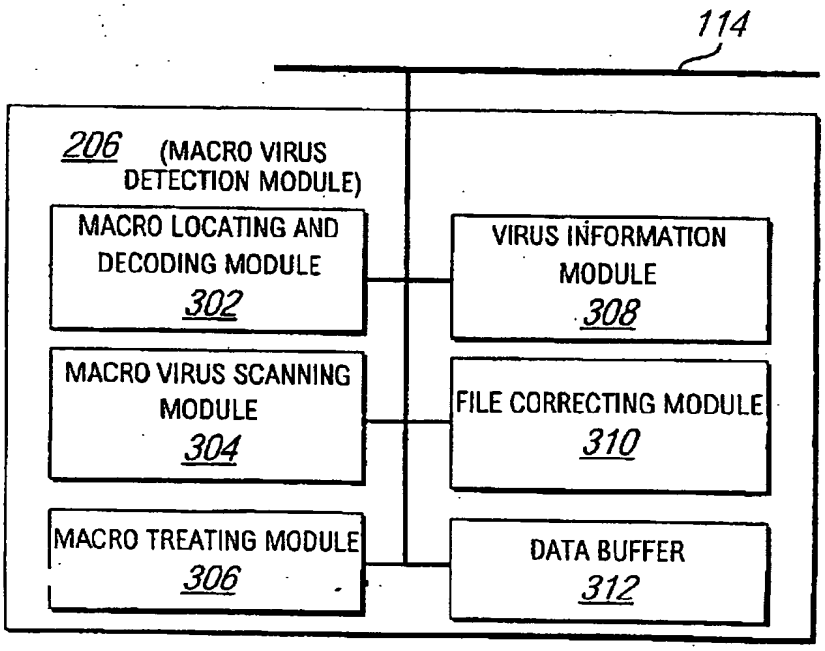
【図２】

FIG. 2



【図３】

FIG. 3



【図４】

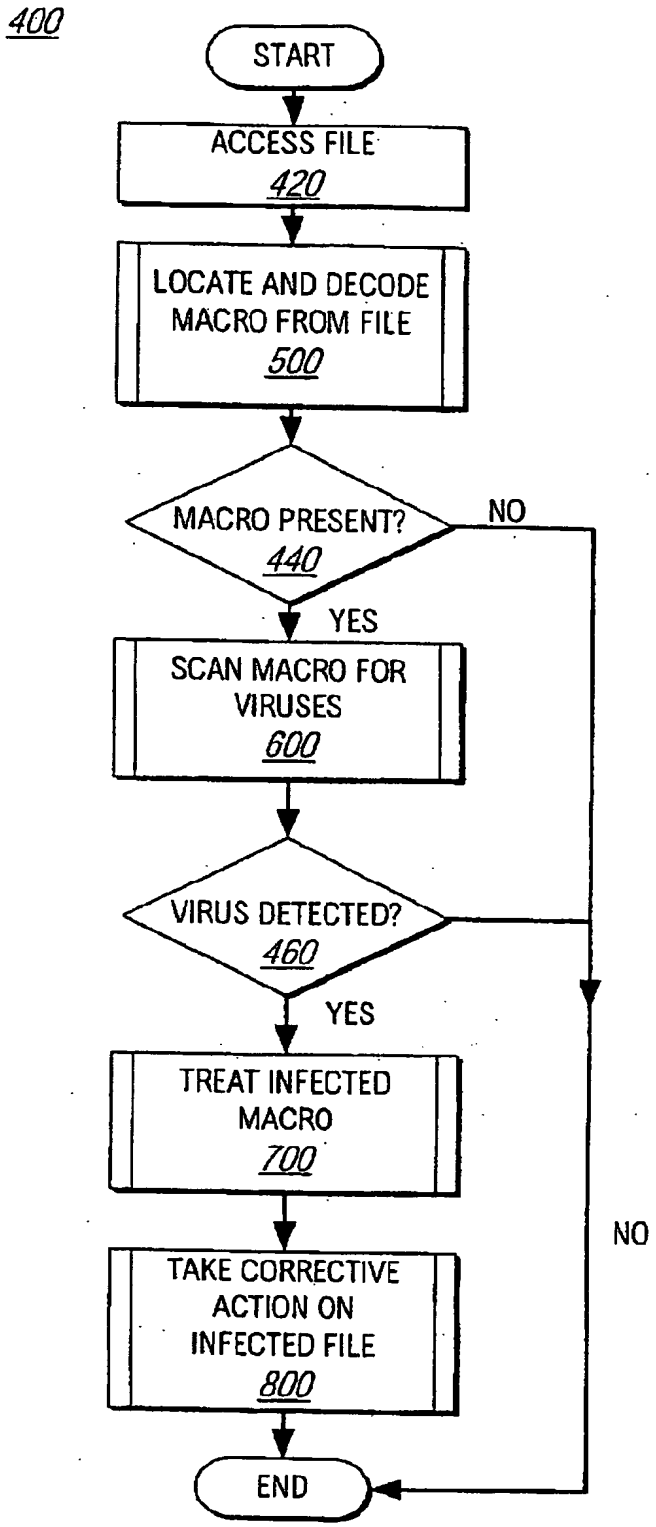

400
START
ACCESS FILE
420
LOCATE AND DECODE MACRO FROM FILE
500
MACRO PRESENT?
440
NO
YES
SCAN MACRO FOR VIRUSES
600
VIRUS DETECTED?
460
YES
TREAT INFECTED MACRO
700
TAKE CORRECTIVE ACTION ON INFECTED FILE
800
NO
END

FIG. 4

【図５】

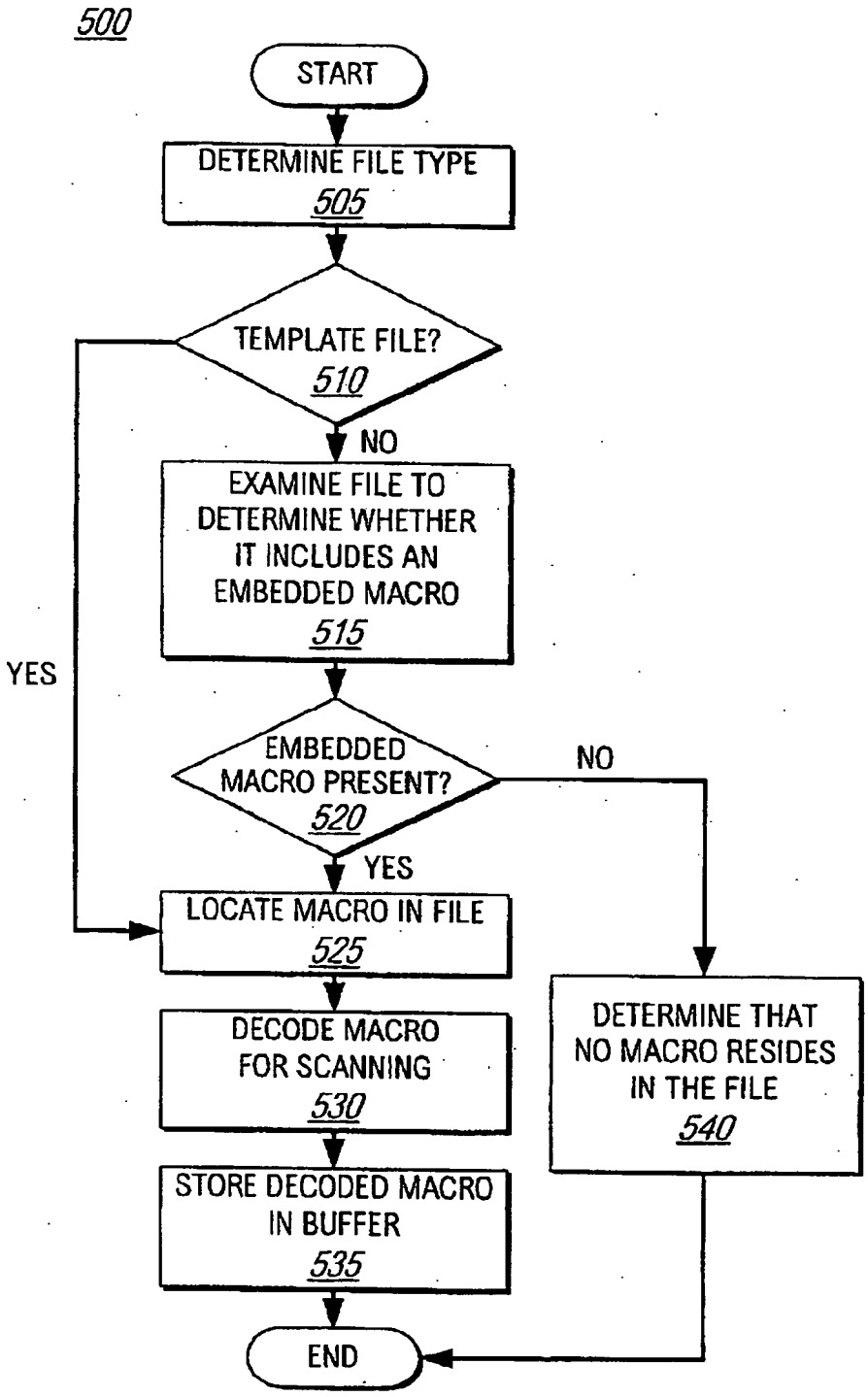

500
START
DETERMINE FILE TYPE
505
TEMPLATE FILE?
510
NO
EXAMINE FILE TO DETERMINE WHETHER IT INCLUDES AN EMBEDDED MACRO
515
YES
EMBEDDED MACRO PRESENT?
520
NO
YES
LOCATE MACRO IN FILE
525
DECODE MACRO FOR SCANNING
530
STORE DECODED MACRO IN BUFFER
535
DETERMINE THAT NO MACRO RESIDES IN THE FILE
540
END

FIG. 5

【図６】

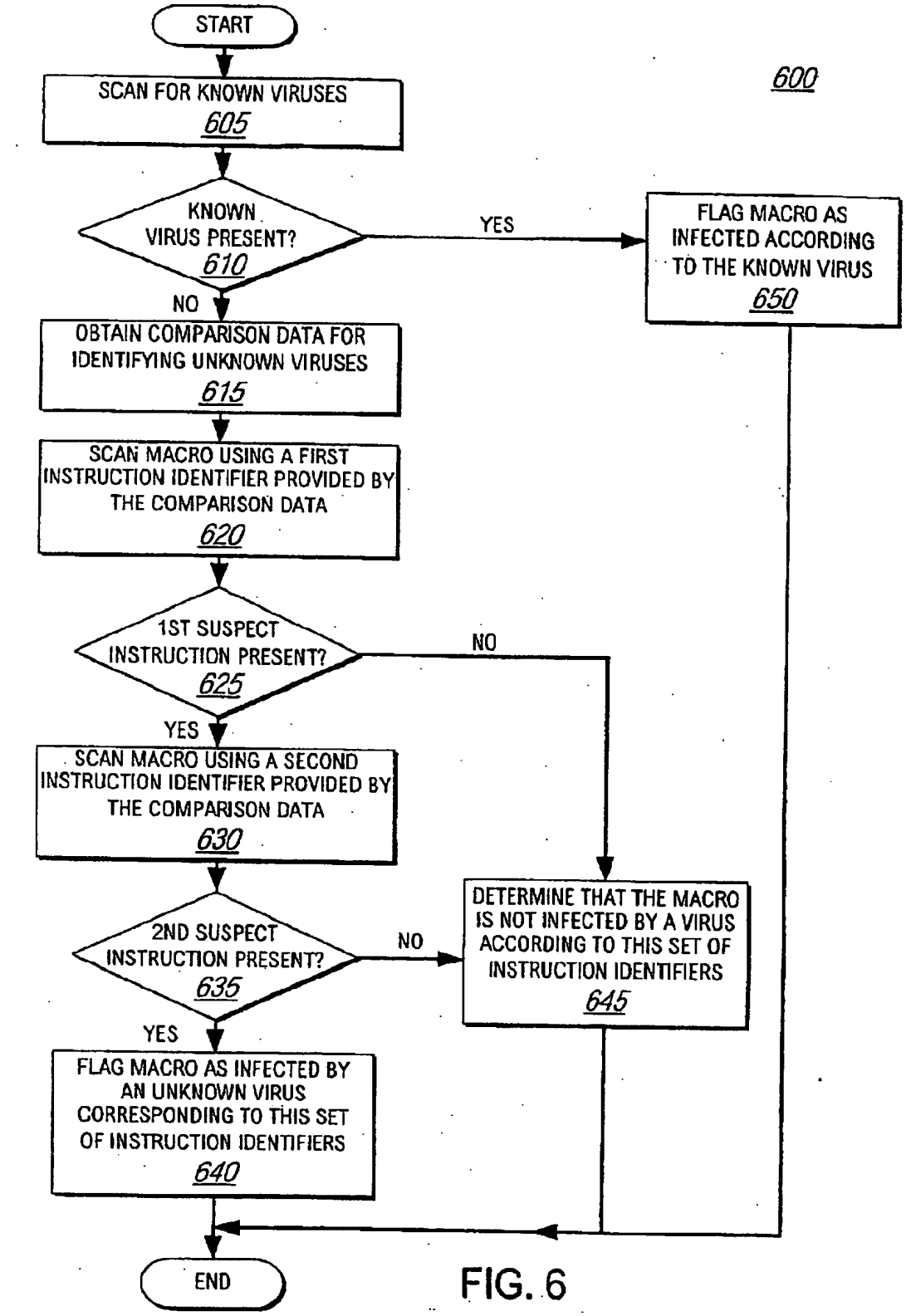

START
SCAN FOR KNOWN VIRUSES
605
600
KNOWN VIRUS PRESENT?
610
YES
FLAG MACRO AS INFECTED ACCORDING TO THE KNOWN VIRUS
650
NO
OBTAIN COMPARISON DATA FOR IDENTIFYING UNKNOWN VIRUSES
615
SCAN MACRO USING A FIRST INSTRUCTION IDENTIFIER PROVIDED BY THE COMPARISON DATA
620
1ST SUSPECT INSTRUCTION PRESENT?
625
NO
YES
SCAN MACRO USING A SECOND INSTRUCTION IDENTIFIER PROVIDED BY THE COMPARISON DATA
630
2ND SUSPECT INSTRUCTION PRESENT?
635
NO
DETERMINE THAT THE MACRO IS NOT INFECTED BY A VIRUS ACCORDING TO THIS SET OF INSTRUCTION IDENTIFIERS
645
YES
FLAG MACRO AS INFECTED BY AN UNKNOWN VIRUS CORRESPONDING TO THIS SET OF INSTRUCTION IDENTIFIERS
640
END

FIG. 6

【図７】

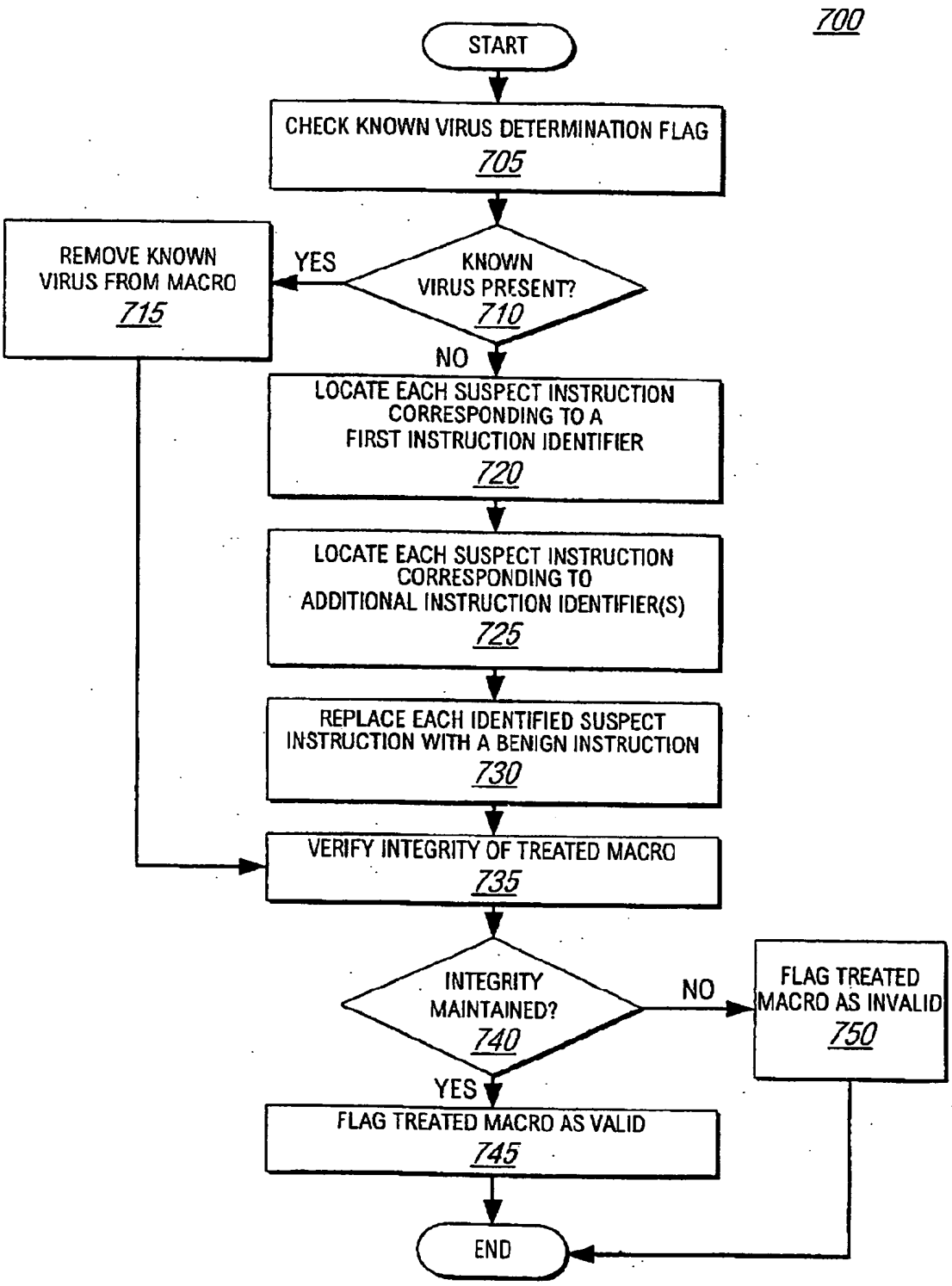

700
START
CHECK KNOWN VIRUS DETERMINATION FLAG
705
KNOWN VIRUS PRESENT?
710
YES
REMOVE KNOWN VIRUS FROM MACRO
715
NO
LOCATE EACH SUSPECT INSTRUCTION CORRESPONDING TO A FIRST INSTRUCTION IDENTIFIER
720
LOCATE EACH SUSPECT INSTRUCTION CORRESPONDING TO ADDITIONAL INSTRUCTION IDENTIFIER(S)
725
REPLACE EACH IDENTIFIED SUSPECT INSTRUCTION WITH A BENIGN INSTRUCTION
730
VERIFY INTEGRITY OF TREATED MACRO
735
INTEGRITY MAINTAINED?
740
NO
FLAG TREATED MACRO AS INVALID
750
YES
FLAG TREATED MACRO AS VALID
745
END

FIG. 7

【図８】

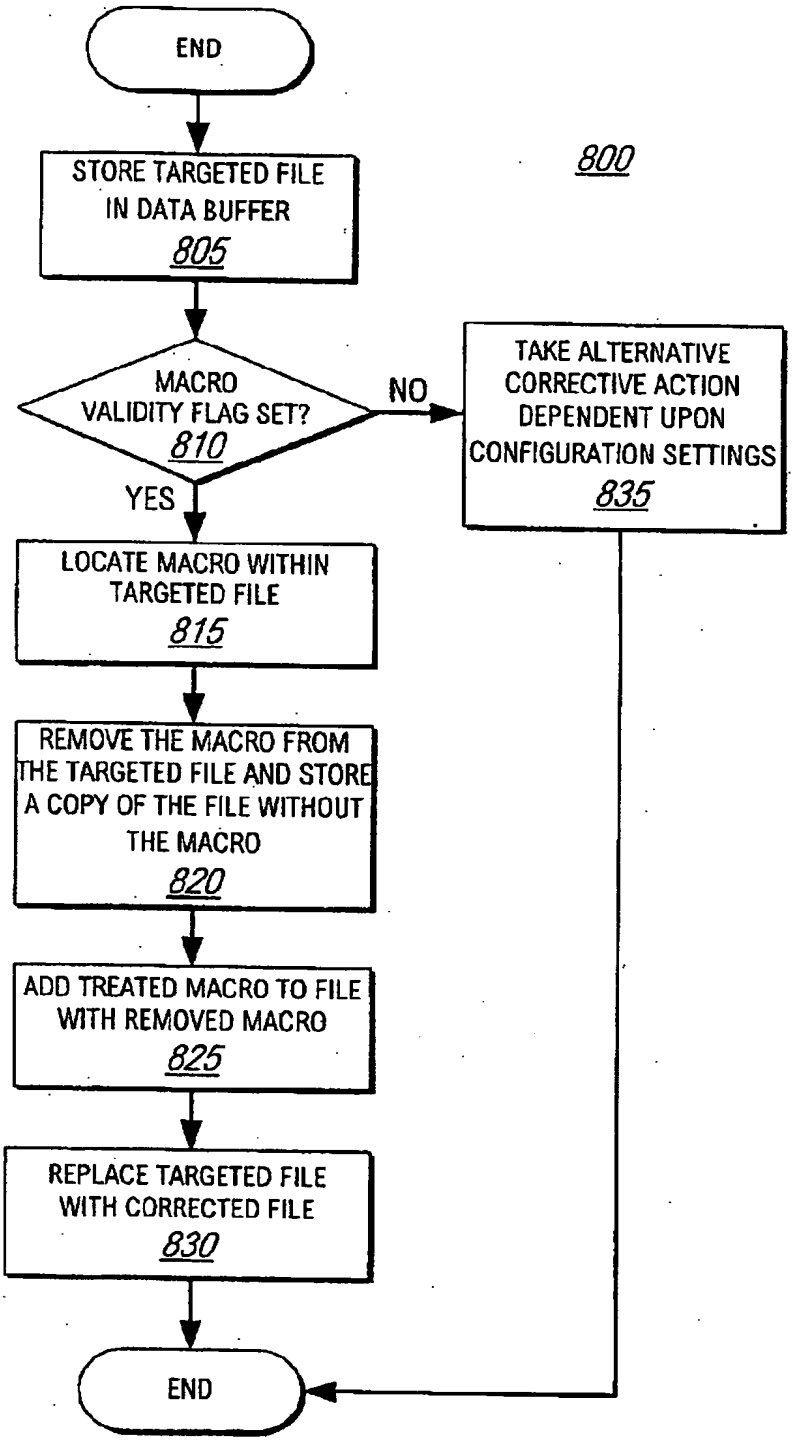

END
STORE TARGETED FILE IN DATA BUFFER
805
800
MACRO VALIDITY FLAG SET?
810
NO
TAKE ALTERNATIVE CORRECTIVE ACTION DEPENDENT UPON CONFIGURATION SETTINGS
835
YES
LOCATE MACRO WITHIN TARGETED FILE
815
REMOVE THE MACRO FROM THE TARGETED FILE AND STORE A COPY OF THE FILE WITHOUT THE MACRO
820
ADD TREATED MACRO TO FILE WITH REMOVED MACRO
825
REPLACE TARGETED FILE WITH CORRECTED FILE
830
END

FIG. 8

【図９】

900

903 組	904 命令ＩＤ番号	905 命令識別子 (TEXT/HEX)
902 1	1	.Format = 1 73 CB 00 0C 6C 01 00
	2	Macro Copy 67 C2 80
902 2	1	.Format = 1 73 CB 00 0C 6C 01 00
	2	Organizer .Copy 64 6F 02 67 DE 00 73 87 02 12 73 7F
3	1	.Format = 1 73 CB 00 0C 6C 01 00
	2	macros. 6D 61 63 72 6F 73 76 08
4	1	FileSaveAs a$,1 12 6C 01 00
	2	MacroCopy 64 67 C2 80 6A 0F 47
5	1	ylformat c: /u" 79 7C 66 6F 72 6D 61 74 20 63 6A
	2	Environ$ ("COMSPEC") 80 05 6A 07 043 4F 4D
. . .	. . .	. . .
902 i	1	...
	2	...
	...	...
	i	...

FIG. 9

【国際調査報告】

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US97/16675

A. CLASSIFICATION OF SUBJECT MATTER
IPC(6) :G06F 11/00
US CL : 385/183.14
According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

U.S. : 385/183.14, 183.13, 183.15, 183.09, 704

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

APS, STN, Internet, IEEE ProQuest
Search terms: macro viruses, virus, software testing or debugging, scanning, signatures

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y,P	BONTCHEV, Possible Macro Virus Attacks and how to Prevent them, Computers & Security. 1996.Vol. 15. No. 7. pages 595-626, specifically pages 599-616.	1-8,10-11,15-19,21,24-26,31-33,35
Y	US 5,440,723 A (ARNOLD et al) 08 August 1995, col. 5, lines 29-68, col. 7, line 34 to col. 10, line 10.	1-8,10-12,15-19,21,24-26,31-33,35
Y	MALARKEY, Comparative Review, Virus Bulletin, May 1996. pages 10-11.	1,15,24,31,35

☐ Further documents are listed in the continuation of Box C. ☐ See patent family annex.

*	Special categories of cited documents:	"T"	later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"A"	document defining the general state of the art which is not considered to be of particular relevance	"X"	document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"E"	earlier document published on or after the international filing date	"Y"	document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"L"	document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	"&"	document member of the same patent family
"O"	document referring to an oral disclosure, use, exhibition or other means		
"P"	document published prior to the international filing date but later than the priority date claimed		

Date of the actual completion of the international search	Date of mailing of the international search report
22 DECEMBER 1997	24 FEB 1998
Name and mailing address of the ISA/US Commissioner of Patents and Trademarks Box PCT Washington, D.C. 20231 Facsimile No. (703) 305-3230	Authorized officer JOSEPH PALYS Telephone No. (703) 305-9685

Form PCT/ISA/210 (second sheet)(July 1992)*

フロントページの続き

(72)発明者 チェン，エヴァ ワイ.
アメリカ合衆国 カリフォルニア州 95014 クペルティーノ，オレンジ アヴェニュー 10408
(72)発明者 ロー，ジョニー ティー.
台湾 シン チュウ カウンティ，チュウ トゥン タウン，クァン ミン ロード，アリー 56，54
(72)発明者 デン，ミン エム.
台湾 台北，ユン ホー ストリート レーン 75，2 フロア ナンバー 5
(72)発明者 チー，レータ エム.
台湾 台北，リン イ ストリート レーン33，1 フロア ナンバー 30

【要約の続き】
を位置特定し、その容疑命令を除去して処置ずみマクロを生ずる。ファイル訂正モジュール（３１０）は、汚染マクロ含有の対象ファイルにアクセスし、マクロ処置モジュール（３０６）からの処置ずみマクロでその汚染マクロを置換する。

* NOTICES *

1.This document has been translated by computer. So the translation may not reflect the original precisely.
2.**** shows the word which can not be translated.
3.In the drawings, any words are not translated.

CLAIMS

[Claim(s)]
1. it is the method of detecting a virus in a broad view in a computer system containing a processor and a storage device -- with a process in which comparison data including information for virus detection is obtained. A method including a process in which a broad view is read, a process in which said broad view is decrypted so that a decrypted broad view may be produced, and a process in which said decrypted broad view is scanned about a virus by comparing said decrypted broad view with said comparison data.
2. Method according to claim 1 of including further process in which said virus is removed from said broad view so that broad view taken a measure may be produced, when process in which said decrypted broad view is scanned shows contamination by said the macroscopic virus.
3. Process in which process in which said broad view is read judges whether object file is template file, A method according to claim 1 including a process in which it is judged whether the object file embeds and a broad view is included when said object file is not a template file, and a process in which position specification of said embedding broad view is carried out when said object file contains a template file.
4. Way according to claim 1 said comparison data contains the 1st suspicion command identifier and the 2nd suspicion command identifier.
A method comprising according to claim 4:
5. A process in which said decrypted data is scanned about said virus, A process in which it is judged whether said decrypted broad view contains the 1st portion corresponding to said 1st suspicion command identifier
A process in which it is judged whether said decrypted broad view contains the 2nd portion corresponding to said 2nd suspicion command identifier.
A process judged as said decrypted broad view including said virus when said decrypted broad view contains said 1st and 2nd portions.

6. Way according to claim 5 said 1st suspicion command identifier detects macro virus enabling-ized command.
7. Way according to claim 6 said 2nd suspicion command identifier detects macro virus duplicate commands.
A method comprising according to claim 2:
8. A process in which said virus is removed, A process in which position specification of the 1st macroinstruction corresponding to said 1st suspicion command identifier is carried out in said decrypted broad view
A process in which said 1st suspicion macroinstruction is removed.

9. Method according to claim 8 of including further process in which said macroscopic absolutely perfect nature with which it dealt is verified, and process in which said contamination broad view in object file is replaced by restored broad view according to said macroscopic absolutely perfect nature verification with which it dealt.
10. A way according to claim 8 a process in which said 1st suspicion macroinstruction is

removed includes a process in which said 1st suspicion command is replaced by non-polluting command.
A method comprising according to claim 8:
11. A process in which said virus is removed, A process in which position specification of the 2nd suspicion macroinstruction corresponding to said 2nd suspicion command identifier is carried out in said decrypted broad view
A process in which said 2nd suspicion macroinstruction is removed from said decrypted broad view so that a broad view taken a measure may be produced.

12. A way according to claim 1 said comparison data contains two or more suspicion command identifier groups.
13. The 1st suspicion command identifier group is string 73 CB 00. 0C6C01 00 and 67C2 A method according to claim 12 containing 80.
14. The 2nd suspicion command identifier group is string 73 CB 00. 0C6C01 00 and 64 6F 02 67 DE00 73 87 01 12 73 7F is included, The 3rd suspicion command identifier group is string 73 CB 00. 0C6C01 00 and 6D61 63 72 6F 7376 08 is included, The 4th suspicion command identifier group is the string 12. 6C01 00 and 64 67C2 80 6A0F 47 are included, The 5th suspicion command identifier group is the string 79. 7C66 6F 72 6D61 74 20 63 6A and 80 05 Six A07 43 A method according to claim 13 containing 4F 4D.
A method characterized by comprising the following of detecting a virus in macro in a computer system containing 15. processor and a storage device.
A macro reading **** process.
A process in which comparison data for detecting a virus is obtained including the 1st suspicion command identifier and the 2nd suspicion command identifier.
A process in which said broad view is scanned so that it may judge whether said broad view contains the 1st portion corresponding to said 1st suspicion command identifier.
A process in which said broad view is scanned so that it may judge whether said broad view contains the 2nd portion corresponding to said 2nd suspicion command identifier, A process judged as said broad view having been polluted with said virus when said broad view contained said 1st and 2nd portions

16. A method according to claim 15 of including further a process in which it deals with said broad view so that a broad view taken a measure may be produced, when judged with said broad view containing said 1st and 2nd portions.
A method comprising according to claim 16:
17. A process in which it deals with said broad view, A process in which position specification of the 1st macroinstruction corresponding to said 1st suspicion command identifier is carried out in said contamination broad view
A process in which said 1st macroinstruction is removed from said contamination broad view so that said contamination broad view may be restored.

A method comprising according to claim 17:
18. A process in which it deals with said broad view, A process in which position specification of the 2nd macroinstruction corresponding to said 2nd suspicion command identifier is carried out in said contamination broad view
A process in which said 2nd macroinstruction is removed from said contamination broad view so that said contamination broad view may be restored.

A method comprising according to claim 15:
19. A process in which said broad view is read, A process in which an object file is accessed
A process in which it is judged whether said object file is a template file.
A process in which it is judged whether the file embeds and a broad view is included when the file is a template file.
A process in which position specification of the embedding broad view is carried out when the

file embeds and a broad view is included.

20. Said 1st suspicion command identifier is string 73 CB 00. 0C6C01 00 is included and said 2nd suspicion command identifier is the string 67C2. A method according to claim 15 containing 80.
21. A way according to claim 15 said comparison data contains two or more suspicion command identifiers.
22. The 1st suspicion command identifier group is string 73 CB 00. 0C6C01 and 67C2 82 is included, The 2nd suspicion command identifier group is string 73 CB00. 0C6C01 00 and 64 6F 02 67 DE 00 73 87 01 12 73 7F is included, The 3rd suspicion command identifier group is string 73 CB 00. 0C6C01 00 and 6D61 63 72 6F 73 76 08 is included, The 4th suspicion command identifier group is the string 12. 6C01 00 and 64 67C2 80 6A 0F 47 are included, The 5th suspicion command identifier group is the string 79. 7C66 6F 72 6D61 74 20 63 6A and 80 05 Six A07 43 A method according to claim 21 containing 4F 4D.
23. A process in which an object file is accessed, and a process in which position specification of said broad view is carried out in said object file, A method according to claim 15 of including further a process in which said broad view is removed from said object file, and a process in which said broad view with which it dealt is added to said object file so that a restored file may be produced.
A device which detects a virus in 24. broad view characterized by comprising the following.
A virus information module which accumulates comparison data for detecting a virus including the 1st suspicion command identifier and the 2nd suspicion command identifier.
A macro virus scanning module which scans said broad view so that it may judge whether said broad view contains the 1st portion corresponding to said 1st suspicion command identifier, and the 2nd portion corresponding to said 2nd suspicion command identifier, while carrying out signal transfer to said virus information module and receiving said comparison data.

25. Carry out signal transfer to said macro virus scanning module, and an object file is accessed, The device according to claim 24 which contains further macro position specification and a decryption module which decrypt the broad view so that it may judge whether the object file is a template file, it may judge whether the object file embeds and a broad view is included and a decrypted broad view may be produced.
26. Signal transfer is carried out to said virus information module, The device according to claim 25 which contains further a macro treatment module which accesses said decrypted broad view, removes the 1st macroinstruction corresponding to said 1st suspicion command identifier, and the 2nd macroinstruction corresponding to said 2nd suspicion command identifier, and produces a broad view taken a measure.
27. Signal transfer is carried out to said macroscopic treatment module, The device according to claim 26 which contains further a file correction module which accesses said object file, removes the broad view for a broad view from an object file of position specification Perilla frutescens (L.) Britton var. crispa (Thunb.) Decne. in said object file, adds said broad view taken a measure to said object file, and produces a corrected file.
28. Said 1st command identifier is string 73 CB 00. 0C6C01 00 is included and said 2nd command identifier is the string 67C2. A method according to claim 27 containing 80.
29. A way according to claim 27 said comparison data contains two or more suspicion command identifier groups.
30. The 1st suspicion command identifier group is string 73 CB 00. 0C6C01 00 and 67C2 82 is included, The 2nd suspicion command identifier group is string 73 CB 00. 0C6C01 00 and 64 6F 02 67 DE 00 7387 01 12 73 7F is included, The 3rd suspicion command identifier group is string 73 CB 00. 0C6C01 00 and 6D61 63 72 6F 73 76 08 is included, The 4th suspicion command identifier group is the string 12. 6C01 00 and 64 67C2 80 6A 0F 47 are included, The 5th suspicion command identifier group is the string 79. 7C66 6F 72 6D61 74 20 63 6A and 80 05 Six A07 43 A method according to claim 29 containing 4F 4D.
A device which detects a virus in 31. broad view characterized by comprising the following.
A means to obtain comparison data containing the 1st suspicion command identifier and the 2nd

suspicion command identifier to virus detection.
A means to scan said broad view so that it may judge whether a broad view contains the 1st portion corresponding to said 1st suspicion command identifier.
A means to scan said broad view so that it may judge whether a broad view contains the 2nd portion corresponding to said 2nd suspicion command identifier.
A means to judge with said broad view being polluted by virus when said 1st and 2nd portions are included.

32. A means which carries out position specification of the said 1st suspicion command identifier and 2nd [said] macroinstructions and 2nd macroinstructions respectively corresponding to a suspicion command identifier in said broad view, [1st] The device according to claim 31 which contains further a means to remove said 1st macroinstruction and said 2nd macroinstruction from said broad view so that a broad view taken a measure may be produced.
33. The device according to claim 32 which contains further a means to judge whether an object file is accessed and the object file contains a broad view.
The device according to claim 33 which contains further a file correction means characterized by comprising the following.
34. A means to access said object file.
A means to remove said broad view from said object file.
A means to add said broad view taken a measure to said object file so that a corrected file may be produced.

A system which detects a virus in 35. broad view characterized by comprising the following.
A storage device which accumulates comparison data for virus detection and a routine containing the 1st suspicion command identifier and the 2nd suspicion command identifier.
A processor which scans said broad view while receiving said comparison data so that it may judge whether signal transfer is carried out to said storage device, and said broad view contains the 1st portion corresponding to said 1st suspicion command identifier, and the 2nd portion corresponding to said 2nd suspicion command identifier.

[Translation done.]

* NOTICES *

1.This document has been translated by computer. So the translation may not reflect the original precisely.
2.**** shows the word which can not be translated.
3.In the drawings, any words are not translated.

DETAILED DESCRIPTION

[Detailed Description of the Invention]
For detection of the virus in macro, and removal A system, a device, and technical field of a background invention of a method invention This invention relates to detection and removal of the virus in computer filing generally.
Explanation of a pertinent art The spread of computer applications and expansion of communication between an immense number of computers made breadth of computer virus remarkably easy, and have promoted. Computer virus is found out by various portions of the code embedded into the computer program. If the program infected by a virus is executed, these code parts will be activated and harmful operation will be produced depending on the case where it does not mean to a computer system.
Detection of a virus is usually performed using the signature scanning technique. The string or signature of the fingerprint EQC usable to virus detection for a known virus became long. In a signature scan, it is investigated whether the file sequence which can be performed is scanned and the extended string who agrees on a known string as a virus is included. If the above-mentioned signature or a string is found out in the file in which the execution is possible, a positive virus judging will be performed. Since it is accompanied by matching with a known pattern, to the virus as which a pattern is not specified, the signature scanning technique is hardly helpful. Especially detection of the kind of strange new virus is completely impossible to the signature scanning technique, and sufficient protection cannot be provided against the mutation virus which takes various shape and forms intentionally in the case of a duplicate. Since the file (for example, file with extension .exe or .com) which can be performed is also usually scanned, the virus which is not in these files is not inspected, therefore it is not detected in a signature scan.
Many application programs are supporting the macroscopic use for automatic execution of a long sequence of operation or a repetitive sequence. A broad view is the command of series, such as menu selection, the bottom of a key press, and a command that is accumulated and receives assignment of a name or a key. With an application program, a broad view answers the call of the push down of a key, or a macro name, and can be started. It is embedded at the application data file and there is also a broad view which stops at the state where it hid from the user. A broad view can be automatically performed without the input from a user.
That is, the broad view which does not need to be known by the user and does not need start up by a user can reside in files, such as an application data file, permanently.
A certain kind of virus resides permanently macroscopically, and performs unexpected harmful operation using a macroinstruction. Those viruses are called macro virus. Since one problem of macro virus does not usually reside in the file which can be performed permanently, it is avoiding the file scanner which can be performed. Since macro virus is hidden into files, such as an application data file, or it may be embedded, detection is escaped. Since the computer user who knows how to use a macro program declinable word word reaches a large number, the number and diversity of macro virus are very large. Therefore, even if it uses the signature scanning technique for detection of the virus in a broad view, since there is much strange macro virus, it is ineffective. Even if it is able to use a comprehensive signature scanner, since generating of

new strange macro virus continues, the scanner obsoletes immediately.
The conventional virus removing technique is also insufficient for the treatment of a virus infection broad view. The technique of these common use searches for a specific known virus, and applies the specific correction technique according to the specific virus detected by the search. There is no effect in the treatment of a virus infection broad view not much for immense **** of macro virus with the strange correction technique. Even if it detects a strange broad view, it does not become an effective solution only by eliminating the file of virus infection macroscopic content. It is because the normal operation which a user wants to hold is included in the virus infection broad view in many cases. Therefore, it is necessary to remove a virus, especially a strange virus from a broad view selectively, and to produce a corrected file without usable infection after that.
Another problem is always changing a virus and information required for detection of these viruses. Therefore, the easy virus detection method and virus detection device of renewal of virus detection information are required. Especially changeable strange macro virus detection information is easily required.
The virus which resides permanently macroscopically needs to be detected as above-mentioned. It is necessary to detect strange macro virus, to defecate macro virus selectively, and to update macro virus detection information simple.
Outline of an invention This invention cancels restrictions of conventional technology and a fault by the system, device, and method of detecting and removing a virus from a broad view.
According to this invention, a macro virus detecting module contains macro position specification and a decryption module, a macro virus scanning module, a macro treatment module, a virus information module, a file correction module, and a data buffer. According to configuration setting out of a macro virus detecting module, one file is set as the object of virus detection, and it copies to a data buffer, and prepares for analysis. The file is examined by macro position specification and a decryption module, and it is judged whether it is a template file. When it judges with a template file, position specification of all the broad views in the template file is carried out, and they are decrypted. When an object file is not a template file, the object file is investigated by macro position specification and a decryption module, and it judges whether an embedding broad view is included, and decrypts by performing the position specification. The decrypted broad view is accumulated in a data buffer.
Signal transfer of the macro scanning module is carried out to this macro position specification, a decryption module, and a data buffer, therefore it accesses a decrypted broad view, and prepares for a virus scan. Signal transfer of the macro virus scanning module is carried out also to a macro virus information module. A macro virus information module includes the information which a macro virus scanning module uses for detection of the known and strange virus in macro. The broad view decrypted [above-mentioned] is first scanned for search of a known virus.
When a known virus is detected, the decrypted broad view is displayed with the flag of infection. The information which relates the decrypted broad view, its flag, and its decrypted broad view with the known broad view detected in macro [the] is accumulated in a data buffer. With a macro treatment module and a file correction module, it deals with the file in which an infection broad view and its infection broad view reside permanently as above-mentioned appropriately, and it is corrected.
When a known virus is not detected, a macro virus scanning module judges whether a virus with a strange decrypted broad view contains. A macro virus scanning module detects strange macro virus using the comparison data accumulated into the virus information module. This comparison data includes the information used for detection of suspicion instruction set doubling in macro. The good example of the group of comparison data contains the 1st and 2nd suspicion command specific codes. They 1st and the 2nd suspicion command judge with the broad view including a virus, as for a macro virus scanning module, when both are contained macroscopically. When a strange virus is detected, according to the group of the suspicion command specific code led to detection of the strange virus, the broad view is indicated by a flag with a contamination broad view. The information led to clear detection as well as the case of detection of a known virus is

accumulated in a data buffer with a contamination broad view, and a macro treatment module and a file correction module deal with a contamination broad view appropriately, and correct it. Since it searches not for the signature of the shape of a specific sequence but for suspicion instruction set doubling, a strange virus is detected with a macro virus scanning module. Since the information for detection of a known virus and a strange virus resides in another module permanently, it is updated easily.

Signal transfer of the macro treatment module is carried out to a macro virus scanning module and a data buffer, and it acquires detection virus-related information by it. A macro treatment module removes a virus from a broad view, generates a defecated broad view or a virus removing finishing broad view, and enables it to repair or correct the file of contamination macroscopic content by a file treatment module. It is judged whether a macro treatment module accesses the decrypted broad view in a data buffer, and has the flag display of contamination by a known virus. When there is a flag display of the contamination according to a known virus macroscopically, the known virus is removed from the broad view. When the broad view is not polluted by the known virus, a macro treatment module deals with a broad view using the group of the command specific code for detection of the strange virus in macro. This macro treatment module is decoded from a macro virus scanning module and a data buffer.

The information about existence of a ** finishing broad view and a virus is received. Position specification of the suspicion command in a decrypted broad view is specified and carried out using a command specific code. next, finishing [a suspicion command is removed from a contamination broad view, and / defecation] by replacing with an uninfected command preferably -- or the broad view [finishing / virus removing] taken a measure is produced. This broad view taken a measure is accumulated in a data buffer, and prepares for access by a file correction module. Finishing treatment [this] macroscopic absolutely perfect nature is inspected, and that validity is indicated by a flag according to an inspection result. When macroscopic absolutely perfect nature is completion of macro treatment and it is maintained, an effective flag indication is given. When absolutely perfect nature is not maintained, it does not indicate by a flag.

Signal transfer of the file correction module is carried out to macro position specification and a decryption module, a macro virus scanning module, a macro treatment module, a data buffer, and a virus information module. The broad view taken a measure and the information about the object file of contamination macroscopic content receive access within a data buffer. A file correction module accesses the object file of the form of a basis, and accumulates the copy of an object file in a data buffer. The copy of an object file contains a contamination broad view. When there is no macro validity flag display, it does not carry out using the broad view taken a measure for contamination macro replacement, The corrective action of substitution, such as object file elimination, is made to perform, and a notice to the user of the existence of a contamination macroscopic content file or removal of the contamination file from an object file, and the object file to a macro-less version are replaced. An object file is corrected when there is a flag display of macro validity, and a file correction module replaces a contamination broad view by the broad view taken a measure. In order to replace a contamination broad view, a file correction module carries out position specification of the contamination broad view, and removes it from an object file, and the version of an object file without [the] a broad view is accumulated in a data buffer. Next, the broad view taken a measure is added to the version of an object file without [said] a broad view, and a corrected file is produced. This corrected file is used for the substitution of an object file (position of a basis). Therefore, a strange virus is removed from a broad view, and the file containing such a broad view is corrected so that a right function may be held.

Brief explanation of the drawings The detailed specific features other than the above of this invention and the above are indicated in detail by the next explanation which referred to the accompanying drawing.

Drawing 1 is a block diagram illustrating the computer system containing the macro virus sensing device by this invention.

Drawing 2 is a block diagram illustrating desirable working example of the storage device by this

invention.
Drawing 3 is a block diagram illustrating desirable working example of the macro virus detecting module by this invention.
Drawing 4 is a flow chart illustrating the macro virus detection and the correcting method by this invention.
Drawing 5 is a flow chart illustrating the macroscopic position specification and decoding method by this invention.
Drawing 6 is a flow chart illustrating the macroscopic scan method for virus search by this invention.
Drawing 7 is a flow chart illustrating the macro treating method by this invention.
Drawing 8 is a flow chart illustrating the file correction method by this invention.
Drawing 9 is a table including the good example of the group of the comparison data used for detection of macro virus.
When the detailed explanatory view 1 of an invention is referred to, the computer system 100 constituted by this invention contains the central processing unit (CPU) 104, the display 102, the storage device 106, the input device 108, the data accumulation device 110, and the communication unit 112. CPU104 is a phon like [in the case of a personal computer]. J. von Neumann It connects with the display 102, the storage device 106, the input device 108, the data accumulation device 110, and the communication unit 112 by bus 114 by the conventional architecture, such as the architecture. Microprocessors, such as Pentium by which CPU104 is marketed from Intel of Santa Clara, California, The display 102 a video monitor and the storage device 106 Random access memory (RAM), As for a keyboard and a mouse, and the data accumulation device 110, it is [the input device 108] preferred to constitute a hard disk drive and the communication unit 112 from devices, such as a modem which makes signal transfer with other systems easy, respectively.
Various computer system configuration other than the above is available, and it is not restrained by any of these composition this invention uses. For example, the processor of the substitution marketed from Motorola can be used for CPU104, and the storage device 106 can also consist of combination of a read only memory (ROM), or RAM and ROM. The system 100 is also connectable with other computer systems by passing a network interface (not shown). Please understand the computer system 100 to be what is not hindered by the mini-computer or a mainframe computer, either.
According to the command constituted by this invention from the memory 106, CPU104, The signal for the macro position specification for judgment of access to computer filing, the judgment of whether these files contain a broad view, and the existence of a strange virus content virus, a macro scan, and the corrective action in the case of virus detection is supplied.
Reference of drawing 2 has shown desirable working example of the storage device 106 constituted according to this invention more to details. The storage device 106 is accumulating the operating system 102, the application program 204, and the macro virus detecting module 206.
As for the operating system 202, it is preferred to be a thing of common use for [, such as WINDOWS 3.1 marketed from Microsoft Corp. of Redmond, Washington,] personal computers, and to constitute. The arbitrary things of various application programs, such as word processing, a spreadsheet, and drawing, can be accumulated in the storage device 106. For example, Microsoft WORD can be accumulated as application for word processing, and Microsoft Excel can be accumulated in the storage device 106 as spreadsheet application, respectively. The application program 204 usually creates an application data file. For example, WORD generates the data file which has ordinary file extension .DOC. The usual application program 204 contains the broad view which makes sequential operation possible without the typematic from a user. Various commands, such as a thing for operation of key press lowering etc. opening, copying and eliminating the thing for simple operation and a file relatively, are included in a conventional broad view. An operating system (or DOS-SHELL) may be called so that a macroinstruction may execute the command of low ranks, such as FORMAT. The macroscopic command to be used usually becomes settled with the application program 204 which supports a macro program

declinable word word. For example, the broad view for a WORD file is written using a WordBasic programming language.
The various operating systems 202, such as OS/2 marketed from IBM, can also be alternatively used for this invention. The various application programs 204 can also be used. Although detection of the macro virus which uses the WordBasic command for a WORD application data file is indicated at a part of working example of this invention, It will be understood by the person skilled in the art that this invention can apply also to the operating system 202 of the above substitution and the substitute application program 204.
The macro virus detecting module 206 contains the routine for correction of the macroscopic treatment and the contamination macroscopic content file which were judged to be the macroscopic scan and virus content for access to a file, the judgment of whether these files contain a broad view, and the judgment of the existence of virus content. The macro virus detecting module 206 cooperates with the operating system 202 and the application program 204, and operates. Although the macro virus detecting module 206 is usually formed into real ** by software, it can carry out [real **]-izing also with hardware or firmware. Although it is preferred that it is different from the operating system 202 and the application program 204 as a graphic display as for the macro virus detecting module 206, A macro virus detecting module can be united with the operating system 202 or the application program 204, and the same virus detection corrective action can be carried out.
When drawing 3 is referred to, desirable working example of the macro virus detecting module 206 contains macro position specification and the decryption module 302, the macro virus scanning module 304, the macro treatment module 306, and the file correction module 310. In addition to these, the comparison data for the treatment of the virus information module 308 of the virus detection in macro and a virus contamination broad view is supplied, and the data buffer 312 accumulates the information for macro virus detection correction. Although illustrated considering the data buffer 312 as a single module including some accumulation positions, two or more individual data buffers can also be used for the various functions of this data buffer 312.
The macro virus detecting module 206 accesses an object file, and judges the existence of macro content. Access to a file is influenced by configuration setting out of the module 302 which the user set up or determined beforehand. For example, a user may be aimed only at a file single for analysis. It can be aimed at file groups, such as a file corresponding to the selected application program 204, or can also be aimed at all the files in the selected directory or a storage region. File analysis can be started with various phenomena. For example, the user can start a virus scan, analysis can be started when [arbitrary] a certain application file is opened, and complete analysis can also be planned for every n boot rises of the system 100, or every specified time interval. As for macro position specification and the decryption module 302, it is preferred to constitute so that the arbitrary files which may contain macro virus are accessed, and access to these files may be performed before starting of an application program (i.e., before opening an application data file). It is because operates with starting of a related application program, therefore some macro virus requires the detection before the scanning start up by a user.
Each of an object file is accessed with the macro virus detecting module 206, and is accumulated in analysis at the data buffer 312. In order to understand easily, analysis of a single file is explained in relation to the specific function of many modules 302, 304, 306, 308, 310, and 312, but this invention can also analyze some files in concurrency or sequentially.
Macro position specification and the decryption module 302 investigate an object file for the judgment of the existence of the judgment of being a macro content type thing, and embedding macroscopic [these files] content, within an object file, carry out position specification of the broad view, and decrypt it.
carrying out signal transfer of this macro position specification and the decryption module 302 to the data buffer 312 -- the analysis to an object file sake -- accessing . A broad view is found out by the template file and embedded at an application data file. Macro position specification and the decryption module 302 judge first whether an object file is a template file. This judgment is performed by checking an extension. For example, if the file is WORD application program 204

relation, a file will be checked about extension .DOT. This .DOT extension shows that a file is a template file.
When not judged with an object file being a template file, the embedding broad view may be included. For example, application data files, such as a WORD file with .DOC extension, may contain the embedding broad view. It is judged whether macro position specification and the decryption module 302 access the object file accumulated in the data buffer 312, the formatting embeds, and a broad view is shown. The formatting field is changed in accordance with the rule of common use of each application program 204, and is supplied by the manufacturer of the application program 204.
A judgment whether an object file contains a broad view, whether the broad view is embedded, whether it has a form of a template file, or it is a file of other forms which can support a broad view will carry out position specification of the broad view in an object file. Signal transfer of macro position specification and the decryption module 302 is carried out to the operating system 202. An operating system contains information shared resources, such as object connection embedding (OLE or OLE2) which is provided by WINDOWS 3.1. This information shared resource provides the details of file structures, such as an application file, and can be made to carry out position specification of the object currently embedded within a file. Although an information shared resource command changes according to the operating system 202, generally it is a simple command which opens an object and which seeks a specific flow, such as writing in by reading to a file. The conventional program technique can be used for working-ization of the information shared resource in macroscopic position specification and decryption. After macroscopic position specification, macro position specification and the decryption module 302 decrypt a broad view, and can be made to perform the scan for virus search. The information shared resource of the operating system 202 is used for the macroscopic decryption to coherent information, and ASCII conversion is used for macroscopic conversion [finishing decryption / to a form suitable for a scan]. A decrypted broad view is accumulated in the data buffer 312. The information which relates a decrypted broad view with the object file of the macroscopic extraction origin is accumulated in the data buffer 312.
Signal transfer of the macro virus scanning module 304 is carried out to macro position specification, the decryption module 302, and the data buffer 312, therefore the module 302 supplies a decrypted broad view to the macro scanning module 304. The desirable method of macro position specification and decryption used for this module 302 is explained still in detail with reference to drawing 5.
The macro virus scanning module 304 contains the routine which scans a decrypted broad view based on comparison with a decrypted broad view and the data from the virus information module 308 for detection of a known virus and a strange virus. The macro virus scanning module 304 can be constituted so that many modes of macro virus detection may be provided. For example, so that a scanning period can be shortened only the thing of a known form only among the specific groups of a virus, i.e., a virus, The composition of being able to constitute so that the thing of a dimorphism type may detect only the thing of a strange form, answering detection of the beginning of a virus, and emitting an alarm, or making the scan of some object files complete before the display of virus detection etc. is also possible.
The macro virus scanning module 304 accesses the decrypted broad view in the data buffer 312, scans the decrypted broad view about a known virus, and when the broad view is not found out, it scans the decrypted broad view about a strange virus. When scanning about a known virus, the macro virus scanning module 304 uses the signature scanning technique. That is, signal transfer of the virus scanning module 304 is carried out to the virus information module 308. The virus information module 308 includes the information which detects a known virus. For example, a virus information module contains the string of data or a signature who specifies a known virus. The virus information module 304 accesses the decrypted broad view in the data buffer 312, scans the decrypted broad view, and judges the existence of virus signature content. A state machine or the same technique can be used for performing this scan. When a known virus signature is found out in macro [the / decrypted], The macro virus scanning module 304 specifies the decrypted broad view with a contamination broad view according to a known virus,

The information which relates the decrypted broad view with the known virus in a data buffer is accumulated, and other modules, such as the macro treatment module 306, enable it to deal with the contamination broad view.
When a known virus is not detected, a macro scanning module scans about the strange virus in a decrypted broad view. The application program 204 includes program language, such as WordBasic including the command which a broad view uses for various operations, in many cases as above-mentioned. Macro virus uses the various operations and commands which perform unnecessary and harmful operation.
The usual application program 204 supports a broad view by providing a template file with a broad view. Word-processing setting out etc. are decorated with a template file, and it includes other setting out. A template file may contain a broad view. Usually, a global template file provides setting out and a broad view for a data file. For example, about Microsoft WORD, global setting out and a macroscopic pool reside in template file NORMAL.DOT permanently. If the application program 204 opens a data file, by it, it opens first, and a global template file will load global setting out and a broad view, and will open a data file after it. The usual data file is formatted so that the application program 204 which does not contain an embedding broad view may be expressed. However, a data file can also be formatted so that it may display that a template file is not included on the application program 204.
A certain kind of macro virus saves the polluted document file in a template format, and a document file or a data file extension (.DOC) is saved, not eliminated. Therefore, a contamination broad view may be embedded into the document of an appearance top application data file. A broad view is contained in the kind of broad view so that a broad view may be performed, when "AutoOpen", "AutoExec", "AutoClose", etc. is opened [a data file]. Therefore, when he does so, he makes an embedding broad view automatically performed, although the user can try to open what is visible to the usual data file. Macro virus also produces a duplicate in other files. For example, copying oneself into a data file and maintaining the usual file extension child to the data file, macro virus has often formatted the data file so that a template format may be displayed.
The file polluted by macro virus will be able to change the format, or is saved by macro virus with the format information whose polluted data file has updated. When the polluted broad view is copied to a global template and other files are opened as a result, it may spread in a file besides them.
The macro virus scanning module 304 contains the high macroinstruction combination of a possibility of being used for macro virus, and the routine which will detect suspicion command combination if it puts in another way. One combination of the suspicion command which the macro virus scanning module 304 detects is the McCloy navel orange-ized command and a macro duplicate command. The McCloy navel orange-ized command is a command which can be set up as formatting of a file displays the macroscopic content file for execution. For example, it can carry out so that a template file may be performed with the application program 204 and a template file may be displayed, when a file is opened [setting out / of file formatting]. Macro duplicate commands are commands which enable the duplicate of macro virus. The combination of the McCloy navel orange-ized command and macro duplicate commands displays macro virus. That is, it is because such a command enables the macroscopic duplicate and execution in a precedence file, and these constitute two usual features of macro virus.
Since it specifies suspicion instruction set doubling, the macro virus scanning module 304 accesses the comparison data from the virus information module 308. This comparison data contains the command identifier group for the specification of suspicion instruction set doubling in a decrypted broad view. The good example of these command identifier group contains the 1st and 2nd suspicion command identifiers. A command identifier is a string of an advance to second base, and a macroscopic scan [finishing decryption] is performed so that it may judge whether these broad views contain the string of these advances to second base, i.e., a suspicion command. When judged with a broad view including suspicion instruction set doubling [which the group of the command identifier defined], the broad view judges with being polluted with the strange virus corresponding to the data group. The macro virus scanning module 304 indicates

the decrypted broad view by a flag with the contamination broad view by a strange virus, The information related with the command identifier which led the decrypted broad view to the strange virus detection in the data buffer 312 is accumulated, and other modules, such as the macro treatment module 306, enable it to deal with a contamination broad view as they thinks best. With reference to drawing 6, it explains further instruction set setting commands of suspicion instruction set doubling [which the macro virus scanning module 304 detected] and, i.e., the formation of macro virus enabling, and a macro virus duplicate etc. in full detail.
As for the virus information module 308, it is preferred to dissociate from other modules 302, 304, 306, 310, and 312 in the macro virus detecting module 206. By it, renewal of the information for macro virus detection becomes easy. For example, the virus information 308 can be updated by copying the new information acquired from media, such as a floppy disk. New information is also downloadable from the Internet resource accessed via the communication unit 112 course of the computer system 100, or the network link (not shown). Protection of the system 100 to the virus which Information Transfer Sub-Division becomes easier with the separated virus information module 308, and updating also becomes easy, therefore includes a strange virus is strengthened.
Signal transfer of the macro treatment module 306 is carried out to the data buffer 312 and the macro virus scanning module 304, therefore it receives the information about detection of the virus in the decrypted broad view from an object file. The routine of the finishing [the macro treatment module 306] decryption [for judging whether the macro virus scanning module 304 detected a known or strange virus in the decrypted broad view] macroscopic check of a state, The routine which removes macro virus from a decrypted broad view, and the routine which verifies finishing treatment macroscopic absolutely perfect nature are included.
The macro treatment module 306 accesses the decrypted broad view in the data buffer 312, and checks a finishing decryption macroscopic state, and it is judged whether the macro virus scanning module 304 detected a known virus. The macroscopic state is expressed with information, including status flags etc., within the data buffer 312. The data buffer 312 is accumulating the information which relates a decrypted broad view with the virus of the known contained macroscopically. This information is supplied from the macro virus scanning module 304 as above-mentioned. The suitable information for the macro virus scanning module 304 can be accumulated, and signal transfer can also be directly carried out to the macro treatment module 306.
When the known virus decision flag is displayed, a macro treatment module is decoded.
The known virus pertinent information for removing the known broad view from a ** finishing broad view is used. Known macro virus is selectively removed from a decrypted broad view, it replaces by non-polluting command, and it is preferred that the portion of the macroscopic emainder holds after it for operation. The broad view taken a measure is accumulated in distinction from the inside of the data buffer 312 with the macro treatment module 306. Next, finishing treatment macroscopic absolutely perfect nature is checked by a macro treatment module, and when absolutely perfect nature is maintained, it indicates to it being effective in the broad view taken a measure by a flag. When finishing treatment macroscopic absolutely perfect nature is not maintained, it is displayed that the broad view taken a measure is invalid. The check of whether a command of the emainder is unhurt and serial connection of a command perform the check of macroscopic absolutely perfect nature by the check of whether to have stopped at the unhurt state. Verification of macroscopic absolutely perfect nature makes it possible to opt for alternative treatment, like other modules, such as the file correction module 310, replace or stop a contamination broad view from an object file on the broad view taken a measure to carry out contamination macroscopic elimination.
When a virus with the strange macro virus scanning module 304 is detected, the macro treatment module 306 deals with the broad view so that the influence of the strange virus may be removed. Like a known virus treatment protocol, when signal transfer of the macro treatment module 306 is carried out to the data buffer 312, and the existence of detection of a strange virus is judged and is judged to be owner **, it specifies the command identifier led to the judgment with a virus. The group of the suspicion command identifier led to detection of the

strange virus in a broad view is used for the macroscopic correction. Each command identifier is related with one or more suspicion commands so that discernment of each command and removal from a broad view may be enabled. The macro treatment module 306 for virus correction decrypts a broad view, and equips correction with it, or accesses a decrypted broad view with the macro virus scanning module 304. It is preferred to remove a suspicion command from a decryption broad view also here, and to replace by non-polluting command. The broad view taken a measure is accumulated in the data buffer 312, and prepares for access by other modules, such as the file correction module 310. Like the above-mentioned virus treatment protocol for known, finishing treatment macroscopic absolutely perfect nature is verified, and an appropriate flag display is stood to the broad view. A suitable macro treatment routine is stated more to details with reference to drawing 7.

Signal transfer of the file correction module 310 is carried out to a module besides the data buffer 312 and the macro treatment module 306, and it receives the information about the virus detected on the broad view from an object file through the signal transfer. The file correction module 310 contains the routine for treatment operation when there is a display of a contamination file. The routine in the macro treatment module 306 can be constituted so that automatic or treatment operations various [time of user permission] may be performed. For example, the file correction module 310 can copy the object file of contamination macroscopic content, can be replaced as and it is macroscopic, and it can replace an object file without a notice to a user as finishing [correction]. [the contamination macroscopic treatment] The file correction module 310 can also be constituted so that the prompt which asks a user the propriety of advance in many stages of a corrective action may be produced. Of course, this operation is performed to a dialogue using the input device 108 and the display 102 of the computer system 100. For example, the file correction module 310 displays on a user that a virus of a certain kind or strange virus was detected in the broad view from an object file. Next, a user receives an inquiry whether it desires to replace an object file by a corrected file. It will be understood by the person skilled in the art that the method of the composition of the file correction module 310 and the method of the PURONTO display in many stages are various.

Signal transfer of the file correction module 310 is carried out to the data buffer 312, and it displays the object file of contamination macroscopic content by it. The file correction module 310 accesses the object file of contamination macroscopic content, and accumulates the copy of the object file in the data buffer 312 for correction of the file. As having mentioned above about macro position specification and the decryption module 302, the macro virus scanning module 304, and the macro treatment module 306, The data buffer 312 is accumulating the information about the relation between an object file and the broad view taken a measure, finishing treatment [the] macroscopic effective invalidity, etc., and the information about the kind of detection virus. The file correction module 310 checks the macro validity flag in the data buffer 312, and it is judged whether maintenance of finishing [of the detection virus by the macro treatment module 306 / removal and treatment] macroscopic absolutely perfect nature was possible. It corrects, when finishing treatment macroscopic absolutely perfect nature is not maintained, and the file correction module 310 replaces the object file in the data buffer 312 as and it is macroscopic. [treatment of the contamination broad view in it] Position specification of the virus contamination broad view is first carried out in a contamination file. Signal transfer with macro position specification and the decryption module 302 can perform this operation, The module 302 and the file correction module 310 which can access the information shared resource of the operating system 202 similarly for macro position specification can also perform independently. The copy of the object file which received contamination is accumulated in the data buffer 312. Next, the broad view taken a measure is added to the version of an object file without [the] a broad view, and a corrected file is produced.

This corrected file is used for the substitution to the object file of the original position. This is the technique of replacing an object file directly by a corrected file. Alternatively, an object file can be eliminated or overwritten and a corrected file can also be accumulated in another position.

As for the broad view corresponding to the object file by which it was indicated by the flag with

invalidity taken a measure, it is preferred not to use for substitution with a contamination broad view. In that case, the file correction module 310 can perform various alternative corrective actions. For example, a user can be told about the object file including a virus, and a contamination broad view can be removed without substitution from a contamination file, or an object file can be removed.

Reference of drawing 4 has shown the method 400 flow chart which detects the strange virus in a broad view. The object file for virus detection is accessed first. An object file is usually in the memory 106, these object files are copied to the data buffer 312, and the macro virus detecting module 206 enables it to detect and remove macro virus from these object files. Although processing is explained in full detail only about one target file, it is possible to access many files in this invention and to inspect. The timing and the range of access are influenced by how this module 206 is constituted as they were mentioned above in explanation of the macro virus detecting module 206. [file] That is, various files can be targeted and virus detection can be started based on selectable conditions to a user. However, as for the macro virus detecting module 206, it is desirable to make it detection and removal of the macro virus which operates without the necessity for starting of the application program 204 by application program 204 starting attained.

After macro position specification and the decryption module 302 accessing an object file and supplying them to a data buffer, they judge the macroscopic existence in the object file, and progress to these macroscopic position specification and decryption. The desirable method 500 of the position specification of a file and decryption is stated more to details with reference to drawing 5. Next, it is judged whether the broad view was found out with position specification and the decryption module 302 (440). When judged with a broad view being in an object file (440), a broad view is scanned with the macro virus scanning module 304 (600), and it is judged whether the broad view is polluted with macro virus. When judged with there being no broad view in an object file (440), this macro virus detecting method is ended. The method (600) of a desirable scan is stated more to details with reference to drawing 6. When a virus is detected in a scan (460), it deals with a contamination broad view by the macro treatment module 306 (700). When a virus is not detected in a scan (460), this virus detection method is ended. The desirable method of macro treatment is stated more to details with reference to drawing 7. A corrective action (800) is performed to the polluted object file after macro treatment (700), and the desirable method of macro virus detection is ended after it. A desirable correcting method is explained in full detail with reference to drawing 8.

Reference of drawing 5 has shown the desirable method 500 of the macro position specification and decryption by this invention. It can be aimed at various files, such as an application data file and a template file. The macro position specification and composite-ized module 302 contains the routine which judges the existence of macro content of an object file first by the judgment (Step 505) of the affiliation kind of the file. This judgment is performed by checking the file extension child of an object file. By the check of an affiliation file type, it can be judged whether an object file is a template file. When the file is a template file, macro position specification and the decryption module 302 do not need to judge the existence of the embedding broad view in an object file. Therefore, when judged with the file being a template file at Step 510, position specification of every broad view in the file is carried out (525), it is decrypted (530), and it prepares for the scan about the virus by the macro virus scanning module 304. Each decrypted broad view is accumulated in the data buffer 312 (535), and prepares for access by the macro virus scanning module 304.

When the file was not a file of the kind which may contain a template file or the other broad view and it is judged at Step 510, an object file is investigated and embedded at Step 515, and macroscopic existence is judged. When there is no embedding broad view, it judges with an object file not containing a broad view at Step 540, and this desirable method is ended. The judgment of whether a file embeds and a broad view is included is performed by checking file formatting. for example, one format -- an application data file -- the display of an extension -- it makes it possible to show that is not concerned with how but a template file is included in the application program 204. When the file format expresses template file content, the object file

may contain an embedding broad view. A file embeds in Step 520, it judges with there being no broad view into an object file in Step 540, when judged with a broad view not being included, and this desirable method 500 is ended.
When the object file did not contain an embedding broad view and it is judged at Step 520, or when it is judged with an object file being a macro content template file at Step 510, position specification of the broad view is carried out at Step 525, and it decrypts at Step 530. Position specification and the decryption module 302 contain the routine using operating system 202 information shared resources, such as object connection embedding (OLE or OLE2) which is provided with WINDOWS 3.1 operating system. As mentioned above about position specification and the decryption module 302 an information shared resource, The command which provides the details about a file structure, i.e., the details of an application file or a template file, is included so that the position specification of an object and decryption which were united with the file may be enabled. The conventional program creation technique can be used for working-ization of the information shared resource for macro position specification and decryption. After carrying out position specification of the broad view and decrypting it, it changes into a binary code (for example, ASCII conversion), and in Step 535, it accumulates in the data buffer 312, and can be made to perform the scan about a virus. Macro position specification and the decryption module 302, The relation between a decrypted broad view and an object file other than accumulation of a decryption broad view is maintained and accumulated, The macro virus scanning module 304, the macro treatment module 306, and the file correction module 310 can be made to perform a right scan and treatment of an object file, and correction. After macroscopic decryption and accumulation are completed at Step 535, the method 500 of this desirable position specification and decryption is ended.
Reference of the flow chart of drawing 6 has shown the desirable method 600 of the virus detection in macro by this invention. The macro virus scanning module 304 accesses the decrypted macro information which position specification and the decryption module 302 provide, and contains the routine which detects existence of a virus by comparison with finishing decryption macroscopic information and the virus information 308.
In the 1st step 605, a decryption broad view is scanned about a known virus. In order to scan about a known virus, the macro virus scanning module 304 uses the signature scanning technique. The macro virus scanning module 304 accesses the decrypted broad view in the data buffer 312, and it is judged whether these broad views contain the virus signature which a virus information module provides. In Step 610, judge whether a decrypted broad view includes the known virus based on the known virus scanning step 605, and in with a known virus, The information which a macro virus scanning module indicates the broad view by a flag with those with contamination by a known virus in Step 615, and associates the decrypted broad view and its known broad view is accumulated in the data buffer 312.
When it judges with a known virus not having been detected at the scanning step 605 at Step 610, the macro virus scanning module 304 scans about the strange virus in a decrypted broad view. In Step 615, the macro virus scanning module 304 takes in a series of command identifiers for detection of a strange virus. The macro scanning module 304 detects the command which is likely to be used for macro virus. These commands are also called a suspicion command. The specific combination of a suspicion command has a high possibility of being used for macro virus. By search of suspicion instruction set doubling, the mistaken virus detection is avoidable. It is because a broad view including two mutually different suspicion commands (or more than it) has a very high possibility of being polluted.
Explanation of the macro virus scanning module 304 described, and the usual application program 204 provides a template file with a broad view. Usually, although an application data file uses a global template file, it can also be formatted so that it may be shown that an embedding broad view is included. For example, a WORD file can be saved in .DOT format for the display of template file content. A contamination document file is made to save in a template format (.DOT), and much macro virus makes a document or a data file extension (.DOC) hold, while it has been eternal. Therefore, a contamination broad view may be embedded on the document which is visible to a mere application data file on appearance. Macro virus makes its own

duplicate to other files. For example, macro virus copies itself to a data file, and it formats a data file into a template format displaying condition in many cases, maintaining the usual file extension child for data files.
One combination of the suspicion command which the macro scanning module 304 detected is the McCloy navel orange-ized command and a macro duplicate command. A macro navel orange-ized command formats a file so that it may mean that the file contains the broad view for execution. For example, file formatting is set up express a template file, and the application program 204 can perform, when a file opens a template file. Macro duplicate commands are commands which enable the duplicate of macro virus. The combination of the McCloy navel orange-ized command and macro duplicate commands expresses macro virus. It is because two the usual features, i.e., the macroscopic duplicate, and execution of the macro virus in a precedence file are attained with these commands.
In specific application files, such as the Microsoft WORD file. If file format field .format is set as one, it is judged as that in which the application program 204 (WORD) embeds at a file, and the broad view is contained, and by suitable start up, a file will be accessed at arbitrary broad views [finishing / embedding], and it will be performed. That is, by setting .format as one in a file, the macroscopic execution in the file is enabled and it is considered that all commands that ask for offer of such setting out by a precedence file are the McCloy navel orange-ized commands. For example, since a command "if dlg.format=0, then dlg.format=1" makes it possible to change .format into 1 from 0 by a precedence file, it is the McCloy navel orange-ized command. The additional copy of the file is saved in another formats, such as a format which indicates that other commands, such as a command "FileSaveAsa$, 1", hold the original file, a file embeds them, and a broad view may be included. Therefore, this kind of command is also the McCloy navel orange-ized command. The various alternative commands which enable macro virus execution in a file will be recognized.
Macro virus duplicate commands are the things of the kind made possible repeatedly [of macro virus]. For example, a command "MacroCopy" copies a broad view, and when the broad view is polluting, it copies all from all harmful command, i.e., transmission, origin to an address. Commands other than the above, such as a command "Organizer.copy", also make macro virus reproduction easy. Please understand that various alternative commands can make macro virus reproduction easy.
After position specification of the macroinstruction from an object file is carried out, it is changed into a binary code for analysis, as the desirable method 500 of macro position specification and decryption was described. A characteristic binary code corresponds also to a suspicion command. For example, a macro virus enabling-ized command "ifdlg.format=0thendlg.format=1" has a specific correspondence binary code like macro virus duplicate commands "MacroCopy." Therefore, the comparison data (615) obtained from the virus information module 308 specifies the 1st and 2nd commands in the broad view from an object file by it, including respectively the characteristic portion of the binary code for the 1st and 2nd commands, or a binary code.
It was judged with a characteristic binary-code portion corresponding to some suspicion commands by this invention. For example, the binary string "73 CB 00 0C 6C 01 00" (hexadecimal notation) corresponds to the command portion ".format=1" found out during some macro virus enabling-ized commands. For example, the above-mentioned command "ifdlg.format=0thendlg.format=1", "ifbewaardlg.format=0thenbewoordlg.format=1; and FileSaveAs.Format=1", And the binary string "73 CB 00 0C 6C 01 00" contains a binary string "FileSaveAs.Name=Filename$(), .Format=1." Therefore, this invention is this 73 CB 00. 0C6C01 Specific strings, such as 00, are used as an identifier for detection of a suspicion macroinstruction different mutually [plurality].
It is preferred to include the command identifier of some groups in the comparison data in the virus information module 308. Various combination of a suspicion command is detectable by use of these command identifier group. Various macro virus enabling-ized commands and macro virus duplicate commands are discriminable using each class of these command identifier. A command identifier is not restricted to a macro virus enabling-ized command and macro virus duplicate

commands. For example, it is usable in the command to which reinitialization is made to carry out without attestation and a command to a computer hard disk device, i.e., the command which changes system construction so that reinitialization without a notice to a user may be made possible, as suspicion command combination.

Reference of drawing 9 has shown the data table including the good example of the command identifier accumulated in the virus information module 308. This good example data table 900 contains the line 902 corresponding to a command identifier different mutually [some]. The text and the correspondence hexadecimal notation 905 of the sequence and command identifier ID number 904 which identify the group of the command identifier 903, and the command identifier binary code are also included in this table. Although it is preferred to include two command identifiers in each class of a command identifier, an additional command identifier may be included in one group. A macro virus judging can be performed based on two or the detection of the other identifier minor group of the three command identifiers. The data table 900 is only illustration. Accumulation of the comparison data to the virus information module 308 can be performed by various techniques.

Reference of the flow chart of drawing 6 will judge whether after obtaining the command identifier of a lot with the macro virus scanning module 304 in Step 615, a decrypted broad view is scanned, and it includes suspicion instruction set doubling [which the command identifier identified]. In Step 620, a decrypted broad view is scanned using the 1st command identifier. For example, string 73CB 00 corresponding to [scan a decrypted broad view (620) and] the 1st command identifier in the 1st group of the command identifier 900 0C6C01 It is judged whether 00 exists or not. A state machine, i.e., the state machine which scan a decrypted broad view and judges the above-mentioned string's existence, performs the scan in Step 620. In Step 625, it is judged whether the 1st suspicion command identifier exists in a decrypted broad view. When it judges with there being no command corresponding to this 1st command identifier, according to this command identifier group, it judges that broad view's un-polluting at (625) and Step 645, and this macro virus scan method 600 is ended.

When it judges with there being a command identifier of the above 1st in Step 625, a decrypted broad view is scanned at Step 630, and the existence of the 2nd command identifier is judged. When it judges with this broad view containing the 2nd suspicion command identifier at Step 635, the decrypted broad view is indicated by a flag at Step 640 with the contamination broad view by the strange virus corresponding to that command identifier group. Decrypted macro correlation ****** is accumulated in the command identifier group led to strange virus detection at the data buffer 312, and other modules, such as the macro treatment module 306, enable it to deal with a contamination broad view as they thinks best.

When it judges with having no 2nd suspicion command identifier at Step 635, the macro virus scanning module 304 judges with having no strange virus the inside of decryption macroscopic according to a command identifier group at Step 645, and the macro virus scan method 600 is ended. This judgment in Step 635 is performed about the single group of a command identifier. Groups other than the above of a command identifier compare a decryption broad view repetitively, and enable the judgment of a strange virus. The existence of the 1st common command identifier can be judged before search of the 2nd various alternative command identifier.

Reference of the flow chart of drawing 7 has shown the suitable contamination macroscopic treating method 700.

In Step 705, it is judged whether the strange virus decision flag was checked and the macro virus scanning module 304 detected a known virus in the decrypted broad view. A known virus decision flag is supplied to the macro treatment module 306 306 in the data buffer 312, i.e., the module which relates a decrypted broad view with the virus information used for detection of a known virus. A known virus is removed from a decrypted broad view using this virus information at Step 715. Removal of the virus from a broad view is performed by replacing a virus by non-polluting commands (no-op etc.). Since a virus is known, it is selectively removable so that a macroscopic normal part may remain as it is. After virus removing, the broad view taken a measure is checked in Step 735, and the absolutely perfect nature is verified. When it judges

with finishing treatment macroscopic absolutely perfect nature being maintained, it indicates to it being effective by a flag with the macro treatment module 306 at Step 745 at the broad view taken a measure. With the macroscopic validity associated data, the broad view taken a measure is accumulated in the data buffer 312, and is maintained at the state.
When judged with the above-mentioned absolutely perfect nature not being maintained in Step 740, in Step 750, the broad view taken a measure is indicated by a flag with invalidity, and pertinent information is similarly accumulated in the data buffer 312.
It returns to drawing 7, and when there was no known virus in a decrypted broad view and the macro treatment module 306 judges, it deals with a broad view so that a strange virus may be removed selectively. The group of the command identifier used for detection of the strange virus in a decrypted broad view can be used for the macro treatment module 306 in the data buffer 312.
The group of this command identifier contains the 1st and 2nd suspicion command identifiers. The 1st suspicion command identifier is used for each position specification of the suspicion command corresponding to the identifier at Step 720. Although position specification of the above-mentioned command is carried out using the technique which was related with detection of the command which the macro virus scanning module 304 uses, and was explained, a decrypted broad view can be scanned. When a command identifier corresponds with the fragment of the command instead of the whole command, each of the fragment in which the macro treatment module 306 was detected is related with the whole command. This correlation is influenced by the macroscopic program declinable word word to be used. The conventional technique can be used for this correlation. An additional suspicion command identifier is used for detection of a suspicion command of correspondence at Step 725. Next, the suspicion command which carried out position specification is replaced at Step 730. It is preferred to replace a suspicion command by non-polluting command as well as a known virus string's substitution. Macroscopic absolutely perfect nature is verified, it indicates by a flag at the broad view taken a measure according to the existence of absolutely perfect nature maintenance, and the macro treating method 700 is ended.
Reference of drawing 8 has shown the desirable correcting method 800 by this invention. The file correction module 310 accesses information, including the object file etc. from which signal transfer was carried out to the data buffer 312 and the various modules 302, 304, 306, 308, and 310, and the detected macro virus and contamination macroscopic content were detected. The object file of macro virus content is accumulated in the data buffer 312 at Step 805. It is preferred to access an object file in the original position and to copy to the data buffer 312 with the contamination broad view. Next, a corrective action is performed by whether the file correction module 310 replaces the broad view in an object file by the broad view taken a measure, or the alternative correction technique is used. A macro validity flag is checked at Step 810, and the finishing treatment macroscopic absolutely perfect nature corresponding to an object file is judged. When it is displayed that the broad view taken a measure is effective, the file correction module 310 replaces the contamination broad view in an object file on the broad view taken a measure. In Step 810, position specification of the contamination broad view is carried out in an object file. This operation is performed using the information shared resource (OLE) of the operating system 202. In Step 820, exploitation of an information shared resource removes the broad view which carried out position specification from an object file, and the version of an object file without [the] a broad view is accumulated in the data buffer 312. In Step 825, the broad view which the macro treatment module 306 generated taken a measure is added to the version of an object file without [the] a broad view, and a corrected file is generated. This corrected file is used for putting in instead of an object file in the original position in Step 830. The object file from the start can also be directly replaced by a corrected file. The object file of correspondence can be eliminated or overwritten and a corrected file can be accumulated in arbitrary positions.
As for this broad view taken a measure, when it returns to Step 810, the broad view corresponding to an object file taken a measure is indicated by a flag with invalidity, and not using for the substitution of a contamination broad view is preferred. That is, in Step 835, a file

correction module performs a substitute corrective action according to a user's configuration setting out. The correction process of substitution, such as performing removal of the contamination file from an object file without substitution, or eliminating a target file, in which a user is told about the correction process of various substitution, i.e., an object file, including a virus will be recognized.

Although this invention has been explained above with reference to specific working example, it will be recognized by the person skilled in the art that various modification is possible. For example, although the series of access, position specification, decryption, detection, and correction has been explained about various modules, when detecting a strange virus in macro, it will be understood that various processes can be incorporated into the usual module which exhibits an equivalent function. It reaches, this invention provides the modification and change of those other than these, and, as for the scope of this invention, above-mentioned working example is limited [these] by only attachment Claim.

A block diagram (drawing 1-3) and a flow chart (drawing 4-8)

The correspondence translation of **** component part (drawing 1)

102 Display 104 central processing unit (CPU)

106 Storage device 108 input-device 110 data-accumulation device 112 communication unit (drawing 2)

202 Operating system 204 application-program 206 Macro virus detecting module (drawing 3)

206 macro virus detecting module 302 macro-position specification and the decryption module 304 macro-virus scanning module 306 -- macro treatment module 308 virus-information module 310 file-correction module 312 data buffer (drawing 4)

420 500 which accesses a file 440 which carries out position specification of the broad view from a file, and decrypts it Those with a broad view?

600 a broad view is scanned about a virus -- 460 virus detection was carried out?

700 800 which deals with a contamination broad view A corrective action is performed to a contamination file (drawing 5).

505 510 which judges a file type Template file?

515 Those with 520 embedding files which investigate a file so that the existence of embedding macroscopic content may be judged?

525 530 which carries out position specification of the broad view in a file 535 which decrypts a broad view for a scan 540 which accumulates a decrypted broad view in a buffer It judges with having no macro permanent residence into a file (drawing 6).

605 610 scanned about a known virus -- those [?] with a known virus

615 625 which scans a broad view using the 1st command identifier from 620 comparison data that incorporates the comparison data for strange virus specific Those [1st] with a suspicion command?

630 635 which scans a broad view using the 2nd command identifier from comparison data Those [2nd] with a suspicion command?

640 645 which indicates by a flag by the strange virus corresponding to this command identifier group at contamination and a broad view -- indicate by a flag macroscopically with those with contamination by the 650 known virus judge that has no macroscopic contamination by this command identifier group (drawing 7)

705 710 which checks a known virus decision flag Those with a known virus?

715 A known virus. 720 removed from a broad view -- each suspicion command corresponding to the 1st command identifier. 730 which carries out position specification of each suspicion command corresponding to the command identifier of the 725 addition which carries out position specification 735 which replaces each specified suspicion command with a non-polluting command 740 which verifies finishing treatment macroscopic absolutely perfect nature Absolutely perfect nature is maintained?

745 750 which indicates the broad view taken a measure to it being effective by a flag The broad view taken a measure is indicated by a flag with invalidity (drawing 8).

805 810 which accumulates an object file in a data buffer Those with a macro validity flag display?

815 825 which removes a broad view from the 820 object files which carry out position specification of the broad view in an object file, and accumulates the duplicate of a macro-less file The 830 object files which add the broad view taken a measure to the file removed [macro] are replaced by a corrected file.

[Translation done.]

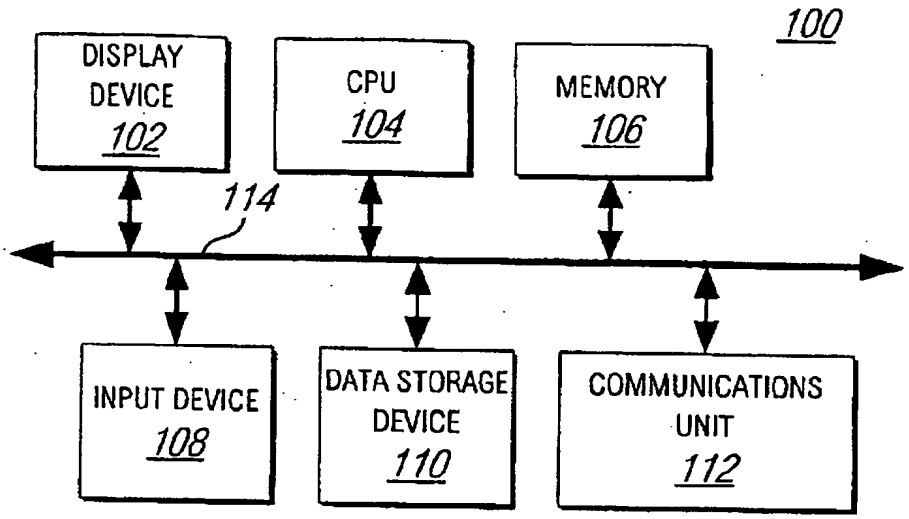
* NOTICES *

1.This document has been translated by computer. So the translation may not reflect the original precisely.
2.**** shows the word which can not be translated.
3.In the drawings, any words are not translated.

DRAWINGS

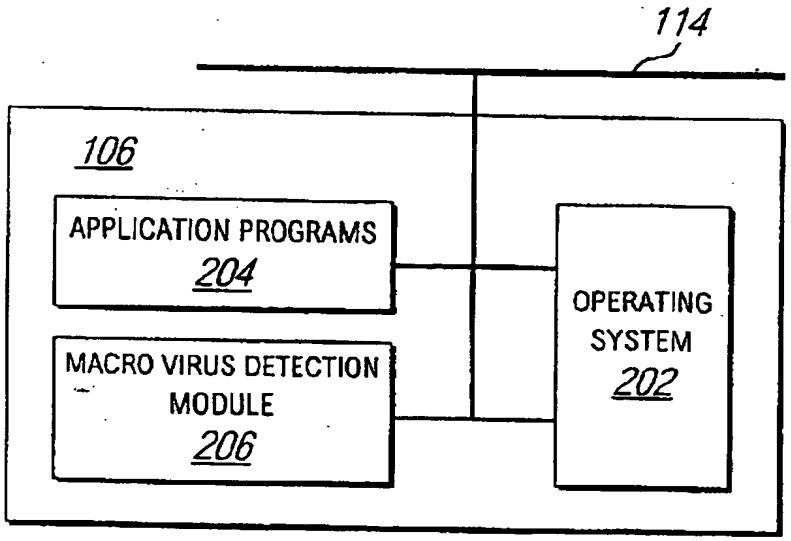
[Drawing 1]

FIG. 1



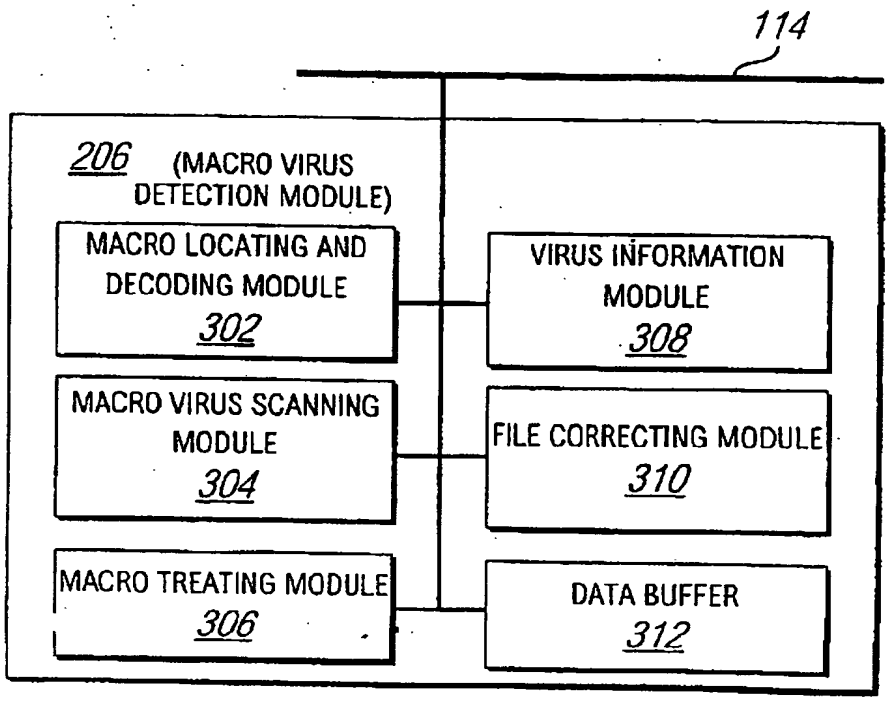
[Drawing 2]

FIG. 2



[Drawing 3]

FIG. 3



[Drawing 4]

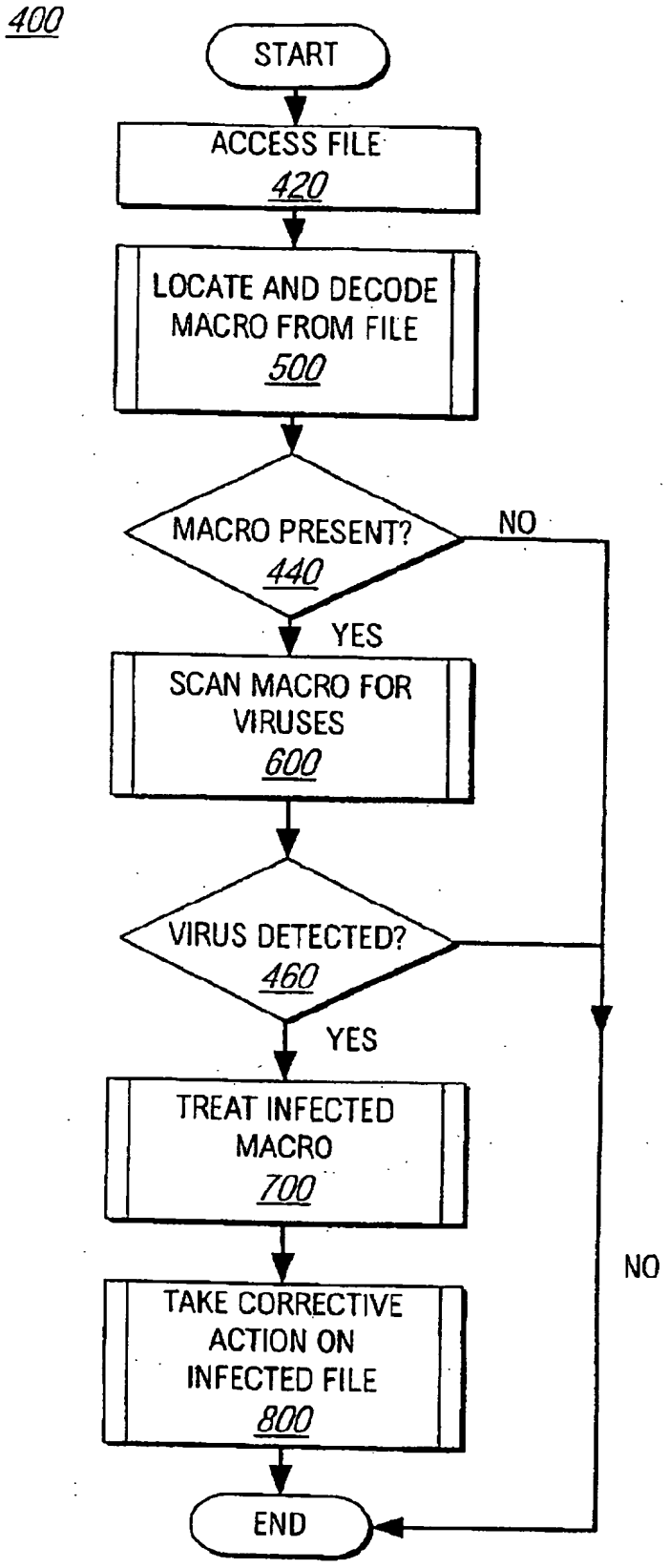


FIG. 4

[Drawing 5]

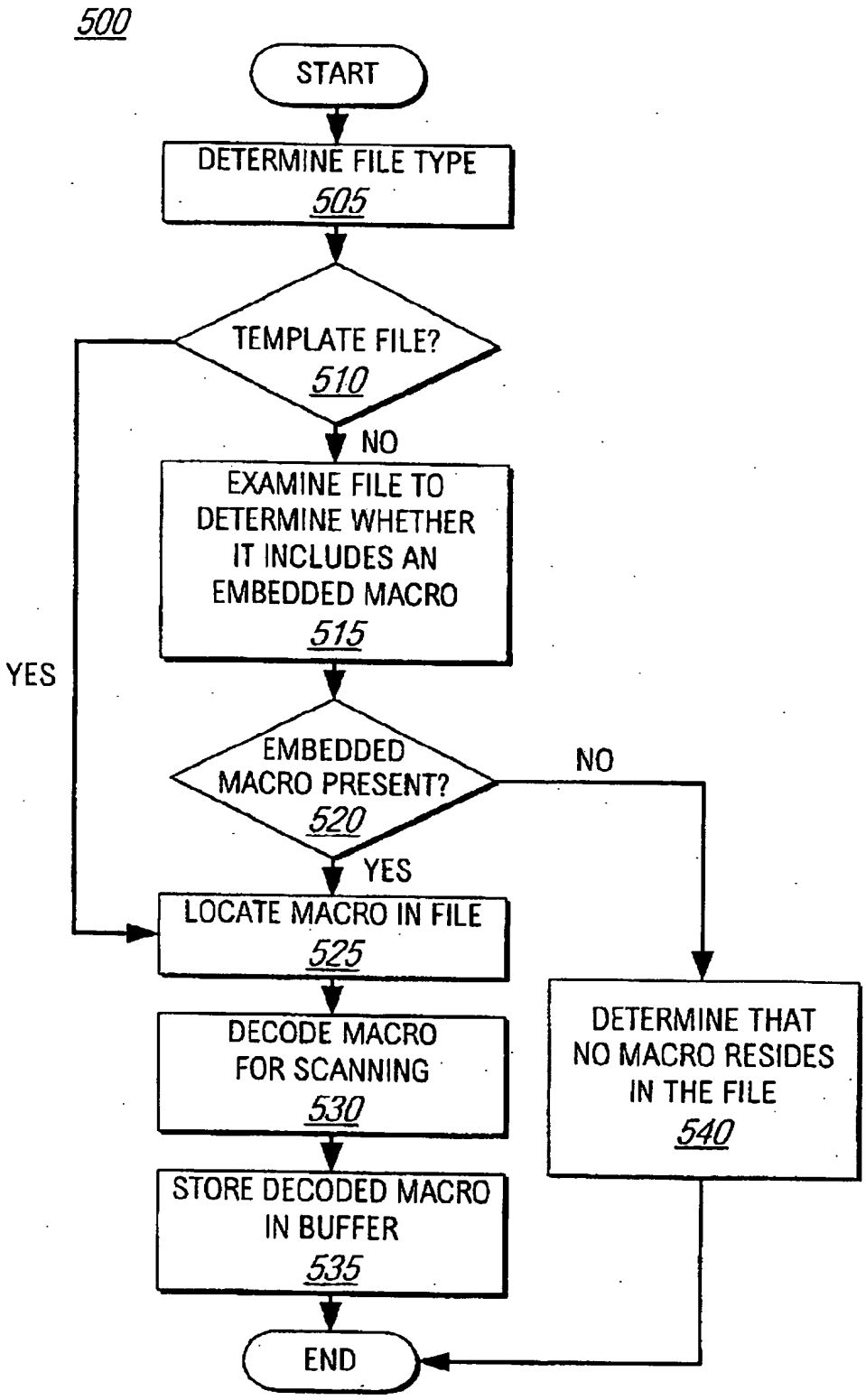


FIG. 5

[Drawing 6]

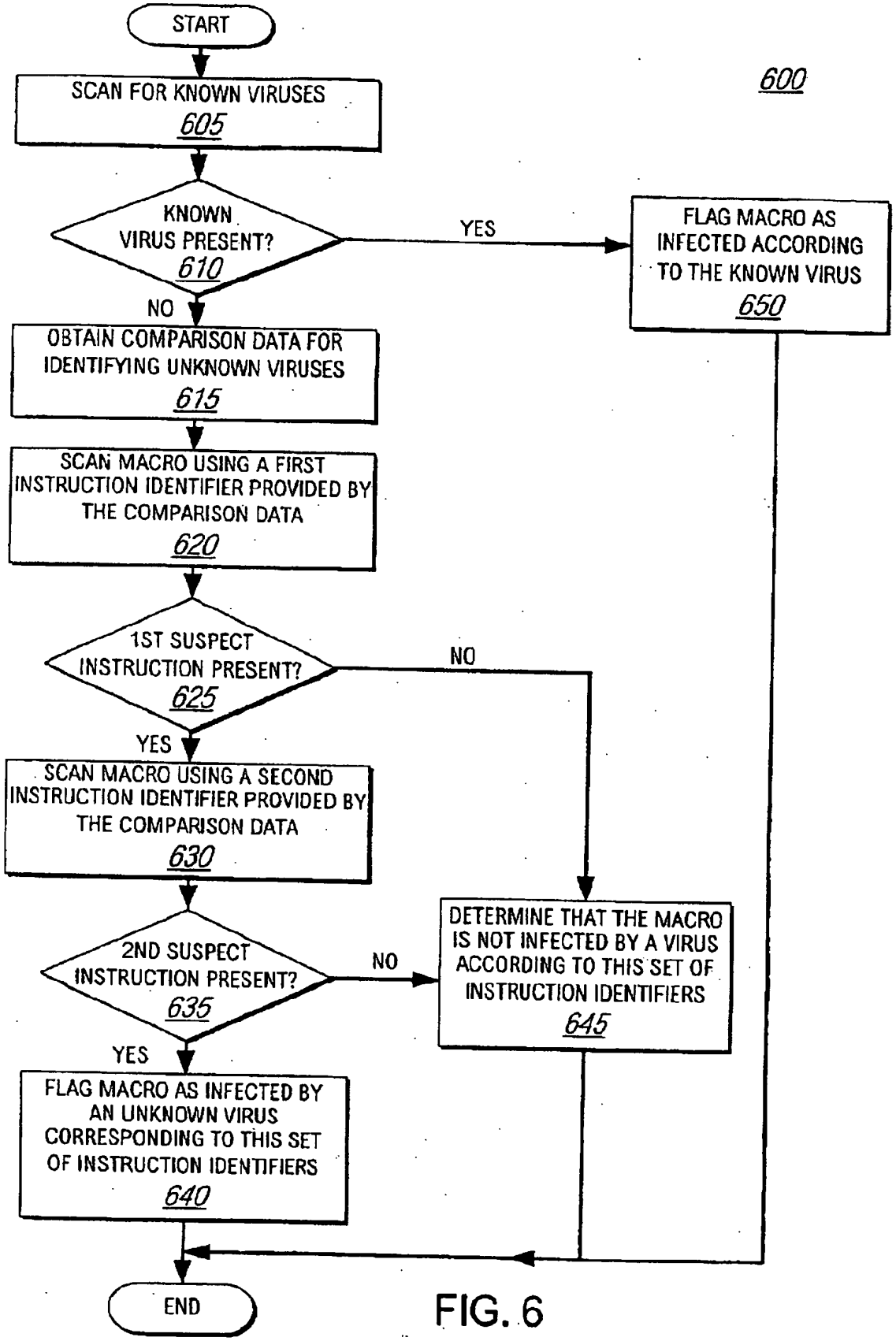


FIG. 6

[Drawing 7]

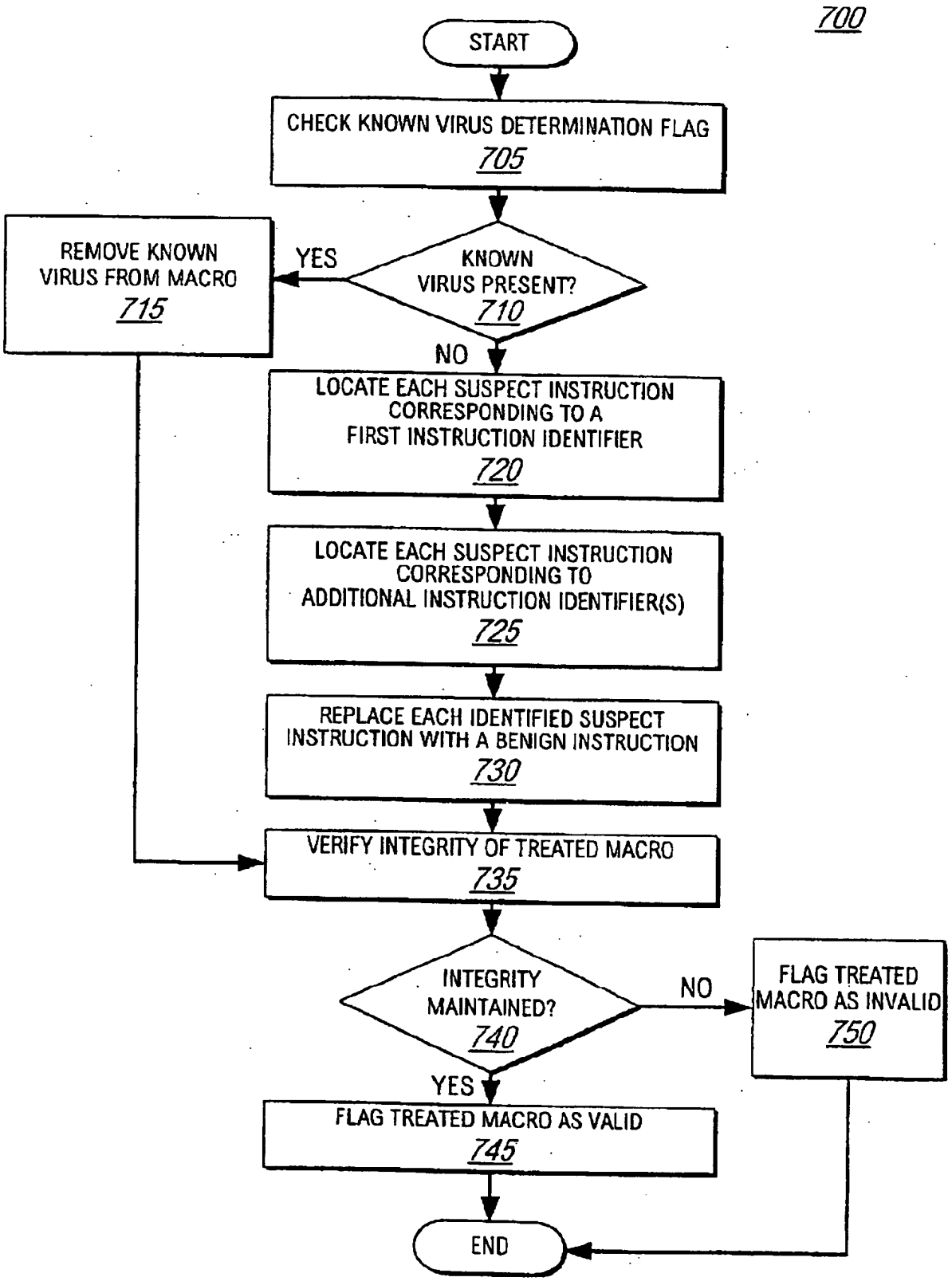


FIG. 7

[Drawing 8]

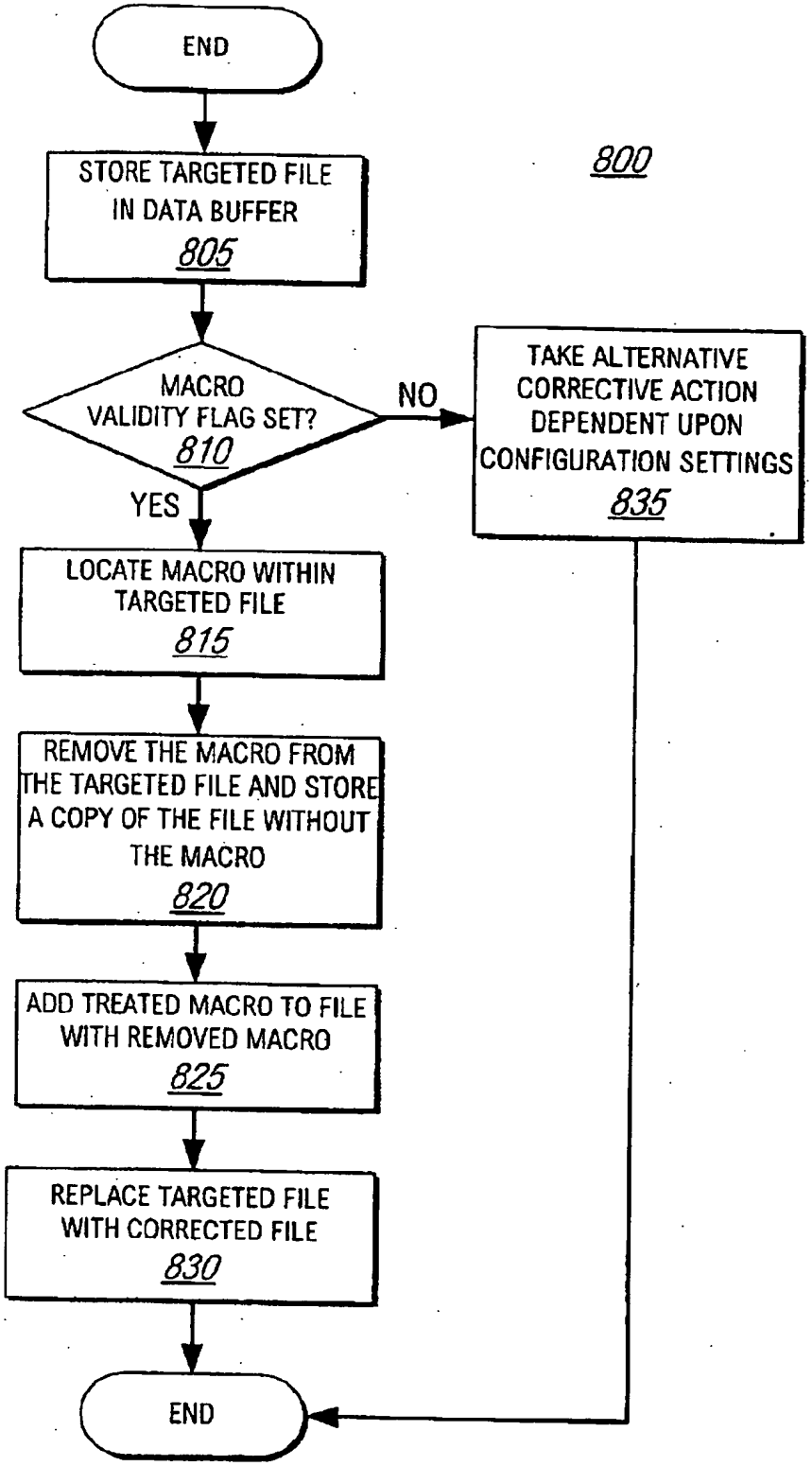


FIG. 8

[Drawing 9]

900

903 組	904 命令ＩＤ番号	905 命令識別子 (TEXT/HEX)
902 1	1	.Format = 1 73 CB 00 0C 6C 01 00
	2	Macro Copy 67 C2 80
902 2	1	.Format = 1 73 CB 00 0C 6C 01 00
	2	Organizer .Copy 64 6F 02 67 DE 00 73 87 02 12 73 7F
3	1	.Format = 1 73 CB 00 0C 6C 01 00
	2	macros. 6D 61 63 72 6F 73 76 08
4	1	FileSaveAs a$,1 12 6C 01 00
	2	MacroCopy 64 67 C2 80 6A 0F 47
5	1	ylformat c: /u" 79 7C 66 6F 72 6D 61 74 20 63 6A
	2	Environ$ ("COMSPEC") 80 05 6A 07 043 4F 4D
. . .	. . .	. . .
902 *i*	1	...
	2	...
	...	...
	j	...

FIG. 9

[Translation done.]